



**ARAÇSAL TASARSIZ AĞLAR İÇİN KONUM TABANLI SEZGİSEL VE
ÖLÇEKLENEBİLİR YÖNLENDİRME PROTOKOLÜ**

Ramazan KOCAOĞLU

**DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

MART 2017

Ramazan KOCAOĞLU tarafından hazırlanan “ARAÇSAL TASARSIZ AĞLAR İÇİN KONUM TABANLI SEZGİSEL VE ÖLÇEKLENEBİLİR YÖNLENDİRME PROTOKOLÜ” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Bilgisayar Mühendisliği Anabilim Dalında DOKTORA TEZİ olarak kabul edilmiştir.

Danışman: Prof. Dr. O. Ayhan ERDEM

Bilgisayar Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

Başkan : Prof. Dr. M. Ali AKCAYOL

Bilgisayar Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

Üye : Doç. Dr. Suat ÖZDEMİR

Bilgisayar Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

Üye : Doç Dr. Süleyman TOSUN

Bilgisayar Mühendisliği, Hacettepe Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

Üye : Doç. Dr. Sevil Şen AKAGÜNDÜZ

Bilgisayar Mühendisliği, Hacettepe Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

Tez Savunma Tarihi: 01/03/2017

Jüri tarafından kabul edilen bu tezin Doktora Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....
Prof. Dr. Hadi GÖKÇEN
Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
 - Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
 - Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
 - Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
 - Bu tezde sunduğum çalışmanın özgün olduğunu,
- bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Ramazan KOCAOĞLU

01/03/2017

ARAÇSAL TASARSIZ AĞLAR İÇİN KONUM TABANLI SEZGİSEL VE ÖLÇEKLENEBİLİR YÖNLENDİRME PROTOKOLÜ

(Doktora Tezi)

Ramazan KOCAOĞLU

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Mart 2017

ÖZET

Bu tezde, araçsal tasarsız ağların karakteristik özellikleri farklı bir bakış açısı ile ele alınmış ve bu ağların kendisine özgü yapısından kaynaklı ortaya çıkan sorunları çözmeye yönelik yeni bir yönlendirme protokolü geliştirilmiştir. Geliştirilen protokol Vehicular Next Position Estimation Routing (VNPER) olarak adlandırılmıştır. Araçların gelecekteki konumlarını tahmin etmeye dayalı olarak geliştirilmiş sezgisel ve ölçeklenebilir bir yönlendirme protokolüdür. İleri görüşlü komşu keşif yaklaşımı ve ileri görüşlü komşuluk algoritması olarak adlandırılan iki yeni yaklaşım öne sürülmüştür. Araçların bir sonraki zaman dilimindeki komşuluk durumlarını öğrenebilmek için ileri görüşlü komşuluk tablosu olarak adlandırılan yeni bir tablo geliştirilmiştir. Ayrıca, rota keşif maliyetini azaltmak için yönlendirme tablosuna ek olarak topoloji tablosu olarak adlandırılan yeni bir tablo geliştirilerek yönlendirme algoritmasına eklenmiştir. VNPER protokolü, araçsal tasarsız ortamlarda daha az paket kaybı, daha az gecikme ve daha fazla miktarda işlem hacmi ile veri iletimine imkan tanımaktadır. Protokol geliştirmek için benzetim aracı olarak Network Simulator 2.34 (NS-2) kullanılmıştır. VNPER protokolü, literatürdeki diğer araçsal ağ yönlendirme protokolleri olan ASTAR, MURU, CAR ve BMFR ile karşılaştırılarak benzetim sonuçları elde edilmiştir. Deneysel sonuçlar, araçsal tasarsız ağlarda VNPER protokolünün paket teslim etme, uçtan uca ortalama gecikme ve toplam veri işlem hacmi metriklerine göre daha başarılı olduğunu göstermiştir. Araç sayısı ve veri miktarının az olduğu ortamlarda VNPER protokolü daha fazla yönlendirme yüküne sahiptir. Ancak, araç sayısı ve veri miktarı artırıldıkça daha az yönlendirme yüküne sahip olmuştur. En az yönlendirme yükü artış oranı VNPER protokolü ile elde edilmiştir. Elde edilen sonuçlar, VNPER protokolünün diğer protokollere göre daha ölçeklenebilir olduğunu göstermektedir.

Bilim Kodu : 92407

Anahtar Kelimeler : VNPER, Araçsal Ağ, Konum tahmini, Topoloji tablosu, NS2

Sayfa Adedi : 116

Danışman : Prof. Dr. O. Ayhan ERDEM

HEURISTIC AND SCALABLE ROUTING PROTOCOL BASED ON PREDICTIVE NEXT POSITION FOR VEHICULAR AD HOC NETWORKS

(Ph. D. Thesis)

Ramazan KOCAOĞLU

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

March 2017

ABSTRACT

In this thesis, Specific attributes of vehicular ad hoc network has been investigated in detail and presented a new routing protocol to overcome the problems caused by unique characters belonging to vehicular network. The developed protocol is called Vehicular Next Position Estimation Routing (VNPER). It is a heuristic and scalable routing protocol based on predicting next positions of vehicles. Two new approaches called discovering future position of neighbor and next location neighbor algorithm have been put forward. A new table called nextneighbor table is developed in order to learn the neighbor position of vehicles in the future. Additionally, a new table called topology table has been developed and added to routing algorithm in order to decrease the cost of route discovering process. VNPER protocol enables data transfer with less packet lose, less delay and more throughput between inter-vehicle communication in vehicular network. To develop routing protocol, Network Simulator 2.34 (ns-2) was used as simulation tool. Simulation results were obtained by comparing VNPER protocol with other vehicular routing protocols such as ASTAR, MURU, CAR and BMFR. Experimental results show that VNPER protocol has obtained more successful performance in terms of packet delivery ratio, average end to end delay and throughput for vehicular ad hoc networks. VNPER protocol has more routing overhead for sparse vehicular network where the number of vehicles and the amount of transferred data packet have less. However, as the number of vehicles and the amount of transferred data are increased, it has less routing overhead for density vehicular network. The least routing overhead ratio was obtained with the developed routing protocol. Therefore, obtained results show that VNPER protocol is more scalable than the other compared vehicular routing protocols.

Science Code : 92407

Key Words : VNPER, Vehicular Network, Position Estimation, Topology table, NS2

Page Number : 116

Supervisor : Prof. Dr. O. Ayhan ERDEM

TEŐEKKÜR

Bu tezin hazırlanma sürecinde ve tüm lisansüstü eğitimim boyunca kendisi ile çalışmaktan gurur duyduğum, tüm çalışmalarımı titizlikle inceleyip bana yol gösteren değerli danışman hocam Prof. Dr. O. Ayhan ERDEM'e, bu süreçte akademik katkı ve desteklerini benden esirgemeyen tez izleme komitesi üyesi kıymetli hocalarım Doç. Dr. Suat ÖZDEMİR ve Doç. Dr. Süleyman TOSUN'a en içten teşekkür ve saygılarımı sunarım.

Doktora tez çalışmam sürecinde desteğini, sevgisini ve özverisini benden hiç esirgemeyen eşim Tuğçe KOCAOĞLU ve yaşama sebepim olacak oğlum Mehmet Kerem KOCAOĞLU'na ithaf ediyorum.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	x
ŞEKİLLERİN LİSTESİ.....	xii
SİMGELER VE KISALTMALAR.....	xvi
1. GİRİŞ.....	1
2. ARAÇSAL TASARSIZ AĞ MİMARİSİ	5
2.1. Araçtan Altyapıya Bağlantı (V2I)	5
2.2. Araçtan Araca Bağlantı (V2V).....	6
2.3. Araçsal Ağların Karakteristik Özellikleri	7
2.3.1. Yüksek hız.....	7
2.3.2. Hareketlilik modeli	8
2.3.3. Ağ yoğunluğu.....	9
2.4. Araçsal Ağ Yönlendirme Protokolleri	9
2.4.1. Hız farkındalıklı V2V yönlendirme protokolleri	10
2.4.2. Hareketlilik tahmini odaklı V2V yönlendirme protokolleri	16
2.4.3. Ağ yoğunluğu odaklı V2V yönlendirme protokolleri.....	22
3. VNPER PROTOKOL TASARIMI	29
3.1. VNPER Protokolünün Paket ve Zamanlayıcı Tipleri.....	29
3.1.1. Rota keşif paketi	29

	Sayfa
3.1.2. Rota teklif paketi	30
3.1.3. Hello paketi yapısı	31
3.1.4. Protokol işleyiş zamanlayıcıları	32
3.2. VNPER Protokolünde Kullanılan Tablolar	34
3.3. VNPER Protokolünün Algoritması	38
3.3.1. VNPER protokolünün paket alım akış şeması	38
3.3.2. VNPER protokolünün rota keşif akış şeması	39
3.3.3. İleri görüşlü komşuluk algoritması	40
3.3.4. VNPER protokolünün rota teklif akış şeması	44
3.3.5. VNPER protokolünün komşu keşif akış şeması	45
3.3.6. VNPER protokolünün ileri görüşlü komşuluk keşif işlemleri	46
4. UYGULAMA ORTAMI VE SENARYO	53
4.1. Kullanılan Benzetim Ortamı	53
4.1.1. NS2 (Network Simulator 2)	53
4.1.2. MOVE (Mobility model generator for vehicular networks)	54
4.1.3. SUMO (Simulation of urban mobility)	56
4.1.4. OSM (Open street map)	57
4.2. Uygulama Ortamı ve Haritası	59
4.2.1. Uygulama ortamının NS2 ortamına aktarılması	61
4.3. Performans Ölçümü için Kullanılan Metrikler	62
4.3.1. Paket teslim etme oranı (PDR)	62
4.3.2. Uçtan uca ortalama gecikme (E2E)	63
4.3.3. Birim zamanda gönderilen veri miktarı (throughput)	65
4.3.4. Yönlendirme yükü	65

	Sayfa
4.4. Uygulama Sonuçları ve Değerlendirme	66
4.4.1. Birinci modelden elde edilen deneysel sonuçlar.....	66
4.4.2. İkinci modelden elde edilen deneysel sonuçlar.....	77
4.4.3. Üçüncü modelden elde edilen deneysel sonuçlar	87
4.4.4. Dördüncü modelden elde edilen deneysel sonuçlar.....	97
5. SONUÇ VE ÖNERİLER.....	109
KAYNAKLAR	111
ÖZGEÇMİŞ	116

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Yüksek hız odaklı V2V yönlendirme protokollerinin karşılaştırılması	16
Çizelge 2.2. Hareketlilik odaklı araçtan araca yönlendirme protokolleri	22
Çizelge 2.3. Araç yoğunluğu odaklı araçtan araca yönlendirme protokolleri.....	27
Çizelge 4.1. Birinci modelde kullanılan parametreler.....	67
Çizelge 4.2. Birinci model birinci senaryoda üretilen araç sayısı ve trafik akışı.....	67
Çizelge 4.3. Birinci model birinci senaryodan elde edilen deneysel sonuçlar.....	68
Çizelge 4.4. Birinci model ikinci senaryoda üretilen araç sayısı ve trafik akışı.....	69
Çizelge 4.5. Birinci model ikinci senaryodan elde edilen deneysel sonuçlar	70
Çizelge 4.6. Birinci model üçüncü senaryoda üretilen araç sayısı ve trafik akışı.....	70
Çizelge 4.7. Birinci model üçüncü senaryodan elde edilen deneysel sonuçlar	71
Çizelge 4.8. Birinci model dördüncü senaryoda üretilen araç sayısı ve trafik akışı.....	71
Çizelge 4.9. Birinci model dördüncü senaryodan elde edilen deneysel sonuçlar	72
Çizelge 4.10. Birinci model beşinci senaryoda üretilen araç sayısı ve trafik akışı	73
Çizelge 4.11. Birinci model beşinci senaryodan elde edilen deneysel sonuçlar	73
Çizelge 4.12. İkinci modelde kullanılan parametreler	78
Çizelge 4.13. İkinci modelde üretilen araç sayısı ve trafik akışı	79
Çizelge 4.14. İkinci model birinci senaryodan elde edilen deneysel sonuçlar	80
Çizelge 4.15. İkinci model ikinci senaryodan elde edilen deneysel sonuçlar.....	80
Çizelge 4.16. İkinci model üçüncü senaryodan elde edilen deneysel sonuçlar	81
Çizelge 4.17. İkinci model dördüncü senaryodan elde edilen deneysel sonuçlar.....	82
Çizelge 4.18. İkinci model beşinci senaryodan elde edilen deneysel sonuçlar	82
Çizelge 4.19. Üçüncü modelde kullanılan parametreler	87
Çizelge 4.20. Üçüncü modelde üretilen araç sayısı ve trafik akışı	88

Çizelge	Sayfa
Çizelge 4.21. Üçüncü model birinci senaryodan elde edilen deneysel sonuçlar	89
Çizelge 4.22. Üçüncü model ikinci senaryodan elde edilen deneysel sonuçlar.....	90
Çizelge 4.23. Üçüncü model üçüncü senaryodan elde edilen deneysel sonuçlar	91
Çizelge 4.24. Üçüncü model dördüncü senaryodan elde edilen deneysel sonuçlar.....	92
Çizelge 4.25. Üçüncü model beşinci senaryodan elde edilen deneysel sonuçlar	93
Çizelge 4.26. Dördüncü modelde kullanılan parametreler.....	97
Çizelge 4.27. Dördüncü modelde üretilen araç sayısı ve trafik akışı.....	98
Çizelge 4.28. Dördüncü model birinci senaryodan elde edilen deneysel sonuçlar	99
Çizelge 4.29. Dördüncü model ikinci senaryodan elde edilen deneysel sonuçlar	100
Çizelge 4.30. Dördüncü model üçüncü senaryodan elde edilen deneysel sonuçlar.....	101
Çizelge 4.31. Dördüncü model dördüncü senaryodan elde edilen deneysel sonuçlar	102
Çizelge 4.32. Dördüncü model beşinci senaryodan elde edilen deneysel sonuçlar	103

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Araçsal tasarsız ağ mimarisi	5
Şekil 2.2. VANET iletişim modelleri	6
Şekil 2.3. Araçtan araca yönlendirme protokollerinin sınıflandırılması.....	9
Şekil 2.4. BMFR protkolünün sıçrama noktası seçimi	12
Şekil 2.5. GpsrJ+ protokolünün sıçrama noktası seçimi.....	18
Şekil 2.6. UMB protokolünün her yöne yayın gönderimi	19
Şekil 2.7. GSR ve SAR protokolü rota keşfi	21
Şekil 2.8. GPCR protokolünün sıçrama noktası seçimi.....	21
Şekil 3.1. RDP paketinin başlık yapısı.....	30
Şekil 3.2. ROP paketinin başlık yapısı.....	31
Şekil 3.3. Hello paketinin başlık yapısı	32
Şekil 3.4. Protokülün işleyişi sırasında kullanılan zamanlayıcılar	33
Şekil 3.5. Komşuluk tablosu ve tabloyu oluşturan nesnelere	35
Şekil 3.6. İleri görüşlü komşuluk tablosu ve tabloyu oluşturan nesnelere	36
Şekil 3.7. Yönlendirme tablosu ve tabloyu oluşturan nesnelere.....	36
Şekil 3.8. Topoloji tablosu ve tabloyu oluşturan nesnelere	37
Şekil 3.9. VNPER protokolünün paket alım akış şeması	39
Şekil 3.10. Rota keşif algoritmasının akış şeması.....	40
Şekil 3.11. RDP paketlerinin yayılımı	40
Şekil 3.12. Ardışık T zamanlarında komşuluk durumları.....	42
Şekil 3.13. Komşuluk güven indeksi	43
Şekil 3.14. Rota teklif paketinin akış şeması	44

Şekil	Sayfa
Şekil 3.15. ROP paketlerinin yayılımı	44
Şekil 3.16. Komşuluk keşif akış şeması.....	45
Şekil 3.17. Hello paketlerinin yayılımı	45
Şekil 3.18. T1 anında düğümlerin ileri görüşlü komşuluk tablosu	46
Şekil 3.19. T1 dinleme anında düğümlerin ileri görüşlü komşuluk tablosu	47
Şekil 3.20. T2 anında düğümlerin ileri görüşlü komşuluk tablosu	47
Şekil 3.21. Koordinat düzeleminde düğümlerin pozisyonları	50
Şekil 3.22. Düğümlerin pozisyon bilgisinin getirilmesi için eklenen kod parçası.....	50
Şekil 3.23. Düğümlerin ivme bilgisinin getirilmesi için eklenen kod parçası	50
Şekil 3.24. Düğümlerin hız bilgisinin getirilmesi için eklenen kod parçası	50
Şekil 3.25. İvme fonksiyonu hesaplaması yapan kod parçası.....	51
Şekil 3.26. VNPER protokolünün algoritmasının akış şeması	52
Şekil 4.1. Linux platformunda MOVE'un çalıştırılması	54
Şekil 4.2. MOVE'un kullanım arayüzü	55
Şekil 4.3. MOVE kullanılarak oluşturulan bir topoloji.....	55
Şekil 4.4. MOVE kullanılarak hazırlanan geniş bir topoloji	56
Şekil 4.5. MOVE'da halihazırda kullanılabilecek topolojiler	56
Şekil 4.6. SUMO'nun genel görüntüsü.....	57
Şekil 4.7. OSM'den alınan Dikimevi bölgesinin haritası	58
Şekil 4.8. SUMO için kullanılabilecek Dikimevi bölgesinin haritası.....	58
Şekil 4.9. Dikimevi bölgesinin NS2 platformunda çalıştırılması	59
Şekil 4.10. Test ortamı için kullanılan gerçek dünya haritası (Kızılay-Sıhhiye).....	60
Şekil 4.11. Test ortamı için kullanılan haritanın SUMO görüntüsü (Kızılay-Sıhhiye).....	60

Şekil	Sayfa
Şekil 4.12. Test ortamından bir görüntü (Kızılay Meydanı).....	61
Şekil 4.13. Kızılay – Sıhhiye meydanı araç hareketlerinin NS2 ortamındaki görüntüsü	61
Şekil 4.14. Kızılay – Sıhhiye meydanı araç hareketlerinin NS2 ortamındaki yakınlaştırılmış görüntüsü	62
Şekil 4.15. Paket teslim etme oranı betiği.....	63
Şekil 4.16. Uçtan uca ortalama gecikme betiği.....	64
Şekil 4.17. Toplam veri işleme hacmi betiği	65
Şekil 4.18. Yönlendirme yükü betiği	66
Şekil 4.19. Birinci model birinci senaryoda araç trafiğinin sumo ortamında oluşturulması	68
Şekil 4.20. Birinci model ikinci senaryoda araç trafiğinin sumo ortamında oluşturulması	69
Şekil 4.21. Birinci model üçüncü senaryoda araç trafiğinin sumo ortamında oluşturulması	70
Şekil 4.22. Birinci model dördüncü senaryoda araç trafiğinin sumo ortamında oluşturulması	72
Şekil 4.23. Birinci model beşinci senaryoda araç trafiğinin sumo ortamında oluşturulması	73
Şekil 4.24. Birinci model için iletilen veri paketi miktarının kıyaslanması.....	75
Şekil 4.25. Birinci model için paket teslim etme oranının kıyaslanması.....	75
Şekil 4.26. Birinci model için uçtan uca gecikme oranının kıyaslanması	76
Şekil 4.27. Birinci model için yönlendirme yükü oranının kıyaslanması.....	76
Şekil 4.28. Birinci model için yönlendirme yükü paketi miktarının kıyaslanması.....	77
Şekil 4.29. Birinci model için işlem hacmi miktarının kıyaslanması	77
Şekil 4.30. İkinci model için araç trafiğinin sumo ortamında oluşturulması.....	79
Şekil 4.31. İkinci model için iletilen veri paketi miktarının kıyaslanması	84
Şekil 4.32. İkinci model için paket teslim etme oranının kıyaslanması.....	84

Şekil	Sayfa
Şekil 4.33. İkinci model için uçtan uca gecikme oranının kıyaslanması	85
Şekil 4.34. İkinci model için yönlendirme yükü oranının kıyaslanması.....	85
Şekil 4.35. İkinci model için üretilen yönlendirme paketi miktarının kıyaslanması	86
Şekil 4.36. İkinci model için işlem hacmi miktarının kıyaslanması	86
Şekil 4.37. Üçüncü model için iletilen veri paketi miktarı kıyaslanması	94
Şekil 4.38. Üçüncü model için paket teslim etme oranının kıyaslanması	94
Şekil 4.39. Üçüncü model için uçtan uca gecikme oranının kıyaslanması	95
Şekil 4.40. Üçüncü model için yönlendirme yükü oranının kıyaslanması	95
Şekil 4.41. Üçüncü model için üretilen yönlendirme paketi miktarının kıyaslanması ...	96
Şekil 4.42. Üçüncü model için işlem hacmi miktarının kıyaslanması.....	96
Şekil 4.43. Dördüncü model için araç trafiğinin sumo ortamında oluşturulması	98
Şekil 4.44. Dördüncü model için iletilen veri paketi miktarının kıyaslanması.....	103
Şekil 4.45. Dördüncü model için paket teslim etme oranının kıyaslanması.....	105
Şekil 4.46. Dördüncü model için uçtan uca gecikme oranının kıyaslanması	105
Şekil 4.47. Dördüncü model için yönlendirme yükü oranının kıyaslanması.....	106
Şekil 4.48. Dördüncü model için üretilen yönlendirme yükü miktarının kıyaslanması .	106
Şekil 4.49. Dördüncü model için işlem hacmi miktarının kıyaslanması	107

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler

Açıklamalar

km	Kilometre
sn	Saniye
GHz	Gigahertz
m	Metre
sa	Saat

Kısaltmalar

Açıklamalar

ASTAR	Anchor-based Street and Traffic Aware Routing - Kanca-tabanlı Cadde ve Trafik Farkındalıklı
AP	Access Point - Kablosuz Erişim Noktası
BMFR	Border node-based Most Forwarding within Radius - Kapsama Alanı Sınır Düğüm-tabanlı Yönlendirme
CAR	Connectivity Aware Routing - Bağlantı Farkındalıklı Yönlendirme
CBDRP	Cluster Based Directional Routing Protocol - Kümeleme tabanlı Yönlü Yönlendirme Protokolü
CBR	Cluster Based Routing - Kümeleme Tabanlı Yönlendirme
CGR	Cached Geocast Routing - Önbellekli Coğrafi Yayın tabanlı Yönlendirme
CMGR	Connectivity Aware Minimum Delay Geographic Routing - Bağlantı Farkındalıklı En Az Gecikmeli Coğrafik Yönlendirme
DECA	Density Aware Reliable Broadcasting -Trafik Yoğunluk Farkındalıklı Güvenilir Yayın

Kısaltmalar**Açıklamalar****D-ODMRP**

Destination driven On Demand Multicast Routing -
Hedef GÜdüllü İstek Tabanlı Çoklu Yayın
Yönlendirme Protokolü

DV-CAST

Distributed Vehicular Broadcast - Dağıtık Araçsal
Yayın

EAEP

Edge Aware Epidemic Protocol - Engel Farkındalıklı
Yaygın Protokol

E2E

End to End Delay - Uçtan Uca Gecikme

GPCR

Greedy Perimeter Coordinator Routing - Anında
Tepkimeli Eş GÜdümlü Yönlendirme

GPGR

Grid-based Predictive Geographical Routing -
Kare Bölgesel Alan tabanlı Öngörülü Coğrafik
Yönlendirme

GPSR

Greedy Perimeter Stateless Routing - Anında
Tepkimeli Çevre Bağımsız Yönlendirme

GPSR+AGF

Advanced Greedy Forwarding - Gelişmiş Anında
Tepkimeli Gönderme

GSR

Geographic Source Routing - Coğrafik Kaynak
tabanlı Yönlendirme

GyTAR

Greedy Traffic-aware Routing -Anında Tepkimeli
Trafik Farkındalıklı Yönlendirme

HyBR

Hybrid Bio-inspired Bee Swarm Routing Protocol -
Arı Kolonisinden Esinlenmeli Yönlendirme Protokolü

HyDi

Hybrid Data Dissemination - Melez Veri Yayımlama

IVG

Inter Vehicle Geocast - Araçlar Arası Coğrafi Yayın

JARR

Junction-based Adaptive Reactive Routing - Kavşak
tabanlı Uyarlamalı Tepkisel Yönlendirme

MANET

Mobile Adhoc Networks - Gezgin Tasarsız Ağlar

MURU

Multi-hop Routing Protocol - Çoklu Sıçrama Tabanlı
Yönlendirme Protokolü

NS-2

Network Simulator 2 - Ağ Benzetim 2

OSM

Open Street Map - Halka Açık Cadde Haritası

Kısaltmalar**Açıklamalar****PASSCAR**

Passive Clustering Aided Routing Protocol - Pasif Kümeleme Yardımlı Yönlendirme Protokolü

PDR

Packet Delivery Ratio - Paket Teslim Etme Oranı

QoS

Quality of Service - Servis Kalitesi

RDP

Route Discovery Packet - Rota Keşif Paketi

RIVER

Reliable Inter-Vehicular Routing Protocol - Araçlar Arası Güvenilir Yönlendirme Protokolü

ROP

Route Offer Packet - Rota Teklif Paketi

ROVER

Robust Vehicular Routing - Sağlam Bağlantılı Araçsal Yönlendirme

RSU

Road Site Unit - Yol Kenar Birimi

SAR

Spatially Aware packet Routing - Uzaysal Boyut Farkındalıklı Paket Yönlendirme

SKVR

Scalable Knowledge-based Vehicular Routing - Ölçeklenebilir Bilgi tabanlı Araçsal Yönlendirme

STAR

Spatial and Traffic Aware Routing - Uzaysal ve Trafik Farkındalıklı Yönlendirme

TIBCRPH

Traffic Infrastructure Based Cluster Routing Protocol with Handoff - Trafik Altyapı Tabanlı Kümelemeli Yönlendirme Protokolü

UMB

Urban Multi-hop Broadcast - Kentsel Çoklu Sıçramalı Yayın

VADD

Vehicle Assisted Data Delivery - Araç Destekli Veri Gönderme

VANET

Vehicular Adhoc Networks - Araçsal Tasarsız Ağlar

VNPER

Vehicular Network Position Estimation Routing - Araçlar için Gelecek Rota Tahmin tabanlı Yönlendirme

V2I

Vehicle to Infrastructure - Araçtan Altyapıya

V2V

Vehicle to Vehicle - Araçtan Araca

1. GİRİŞ

Son yıllarda iletişim alanında yaşanan gelişmeler sonucunda birçok farklı kavram ve teknoloji ortaya çıkmıştır. Özellikle IPv6 ve nesnelerin İnterneti gibi teknolojilerin ortaya çıkması, haberleşme alanında çığır açacak birçok yeniliğin insanların günlük yaşantısına girmesine neden olacaktır. Araçsal tasarsız ağların (Vehicular Adhoc Network-VANET) gelecekte kullanılacak olan iletişim teknolojilerinden biri olacağı tahmin edilmektedir. VANET'ler hareketli tasarsız ağların (Mobile Adhoc Network-MANET) bir türü olarak ortaya çıkmıştır. VANET kısaca, otoyol ve kentsel alanlarda seyir halinde olan araçların çeşitli amaçlara ulaşmak için oluşturdukları ortak ağ yapıları olarak tanımlanabilir. Son yıllarda yaşanan teknolojik gelişmelerin neticesinde bu alanda çalışmalar hızlanmıştır. VANET'ler başta trafik akışının düzenlenmesi, trafik kazalarına anlık müdahale edilebilmesi, acil durumlarda kritik bölgelerden bilgi toplanmasına yönelik işlevleri yerine getirebilmek için geliştirilmiş ağlardır. VANET'lerin en büyük avantajı mevcut bir ağ altyapısına ihtiyaç duymaksızın acil ve olağanüstü durumlarda alternatif bir iletişim aracı olarak da kullanılabilmesidir. Halihazırda sabit bir ağ altyapısına ihtiyaç duymadığı gibi farklı amaçlara hizmet edebilmek ve VANET'in performansını artırabilmek için hücresele ağ altyapısını veya VANET iletişimde kullanılmak için tasarlanan yol kenarı birimlerini (Road Site Unit-RSU) kullanabilir [1-4].

VANET'lerde her bir araç hem veri alımı hem de veri gönderimi yapabilen, ağ içerisindeki bir düğüm olarak değerlendirilir. Gelecekte her bir aracın üretimi sırasında araçlara VANET iletişimi için bir donanım ekleneceği ve tüm araçlarda olmasının zorunlu hale getirileceği tahmin edilmektedir. VANET'lerin günlük hayatımızda birçok amaç için kullanılabileceği düşünülmektedir. VANET'lerin etkin ve güvenli şekilde kullanılabilmesi birçok faktörün bir arada uyum içinde çalışabilmesine bağlıdır. Araçların VANET ortamına bağlanmasını sağlayan kablosuz erişim cihazının (Access Point-AP) donanımsal özelliklerinin ve mimari yapısının mevcut kablosuz erişim cihazlarından hem donanım hem de çalışma mimarisi açısından tamamen farklı bir yapıya sahip olması gerekmektedir. Her bir araçta bulunan kablosuz erişim cihazlarından elde edilen veriler diğer araçlar üzerinden aktarılarak belirli bir hedefe ulaştırılır. 802.11p araştırma grubu VANET ortamında kullanılacak olan kablosuz erişim standartlarının belirlenmesi üzerine çalışmalar yapmaktadır [1-3]. Yüksek hızlardaki araçların birbirleriyle ve RSU birimleriyle sorunsuz bir şekilde veri iletişimi yapılabilmesi

için gerekli olan ihtiyaçları karşılamaya yönelik çalışmalardan sorumlu olan IEEE 802.11 grubudur. VANET iletişimi, 5.9 Ghz bant genişliği frekansında çalışacak şekilde IEEE 802.11p araştırma grubu tarafından lisanslanmış ve standart hale getirilmiştir [1-4].

VANET'lerin yüksek hızlara sahip olması ve hızlı yer değişimli yapısından dolayı düğümler arasında bağlantı kopmalarına neden olmaktadır. VANET'lerde yapılan çalışmalar daha çok yüksek hızın ortaya çıkardığı sorunların çözümüne yöneliktir. VANET'ler için iyi bir yönlendirme protokolünün tasarımı, kendisine özgü olan aşırı hız, sıklıkla topoloji değişikliği gibi özellikleri içinde barındıran bir algoritmaya sahip olmasına bağlıdır. Bundan dolayı tasarsız ağlar için kullanılan iletişim protokolleri VANET'lerde kullanışsız olmaktadır. VANET'lerdeki yönlendirme protokollerine yönelik çalışmaların büyük bir kısmında araç yoğunluğunun sabit olduğu varsayılmıştır. Ancak, araç yoğunluğu bölgeden bölgeye değiştiği gibi, şehir merkezinden otoyollara göre de değişiklik göstermektedir. Şehir merkezlerinde yoğunluk fazla iken, otoyollarda yoğunluk çok daha azdır. Hatta merkezlerdeki yoğunluk seviyesi bölgelere göre de değişkenlik gösterir. Yoğunluk bölgeden bölgeye değişebileceği gibi zamana göre de değişkenlik göstermektedir. Gündüz vakti mesai başlangıç ve bitiş zamanlarında şehir içinde trafik yoğunluğu geceye oranla çok daha fazla olmaktadır. VANET'lerde ağ yoğunluğunun fazla olması broadcast storm (aşırı miktarda yayın mesajı) sorununu ortaya çıkarırken, seyrek olan ağlarda ise bağlantı kopmalarından kaynaklı problemler ortaya çıkmaktadır. Bu durumda veri paketleri uygun bir bağlantı kurulana kadar bellekte bekletilmektedir. Bağlantı oluşturulması sonrasında veriler hedefine doğru iletmeye başlanır. Literatürde bu yaklaşımı temel alan çalışmalar bulunmaktadır [1-3].

Araçların hareketleri rastgele olmamasına rağmen, hareket yönlerinin önceden doğru şekilde tahmin edilebilmesi mümkün değildir. Ancak önceden belirlenmiş yollar üzerinde hareket ettiklerinden, hareketleri de sınırlıdır ve belirli bir hareket modeline uygunluk gösterir. Araç hareketlerinin yollara bağımlı olmasının getirdiği topolojik yapı, gelecekte yapılacak olan çalışmalara da yön verebilir. Yollara bağlı bir topolojik yapının olması ve araçların tamamen rastgele hareket durumunun olmaması araçsal ağlar için bir avantaj olarak kabul edilir. Yüksek hızın getirdiği hareketlilik ise sürekli olarak topolojisinin değişmesine neden olmaktadır. Bu durum araçsal tasarsız ağlar için bir dezavantajdır. Araç yoğunluğunu seyrek ya da yoğun oluşu ise farklı sorunları beraberinde getirmektedir.

Bu tez kapsamında araçsal ağlarda kullanılmak amacıyla bir yönlendirme protokolü geliştirilmiştir. Geliştirilen protokol VNPER olarak adlandırılmıştır. Araçsal ağların kendisine özgü yapısından kaynaklı ortaya çıkan sorunları çözmek amacıyla geliştirilmiştir. VNPER protokolü, NS2 benzetim ortamı kullanılarak geliştirilmiştir. Move ve SUMO programları NS2 ile beraber tümleşik şekilde kullanılabilir hale getirilerek gerçek dünya ortamlarında test yapılabilmesi sağlanmıştır. Araç hareket şekilleri ve trafik modellemesi Move ve SUMO ile yapılmıştır. Gerçek dünya haritaları OSM kullanılarak benzetim ortamına aktarılmıştır. VNPER protokolü, literatürde bulunan farklı araçsal ağ protokolleri ile kıyaslanarak performans ölçümleri karşılaştırmalı olarak sunulmuştur.

VNPER protokolü, araçların bir sonraki zaman dilimindeki konumlarını tahmin ederek çalışmayı temel alan bir protokoldür. Bu işlemi yerine getirebilmek için ileri görüşlü komşuluk algoritması ve ileri görüşlü komşu keşif yaklaşımı olarak adlandırılan iki yeni yöntem geliştirilmiştir. Buna ek olarak yönlendirme algoritması içerisinde yedek rotaların saklandığı topoloji tablosu olarak adlandırılan yeni bir tablo geliştirilmiştir. Bu tablo sayesinde, rota keşif işlemlerinin ortaya çıkardığı yönlendirme maliyeti azaltılarak VNPER protokolünün ölçeklenebilir bir yapıya sahip olması sağlanmıştır. İleri görüşlü komşu keşif yaklaşımı sayesinde araçların bir sonraki zaman diliminde sahip oldukları komşuluklar hesaplanmaktadır. İleri görüşlü komşuluk algoritması sayesinde ise, bir sonraki zaman dilimindeki komşuluk dizilimleri kullanılarak yönlendirme ve topoloji tablosu oluşturulmaktadır. Bu sayede bir sonraki zaman dilimindeki araç dizilimlerine göre rota hesabı ve rota tahmini yapılabilir. Sonuç olarak rota bozulmaları önceden tahmin edilerek paket kaybına neden olacak rotaların seçilmesi engellenmektedir. Elde edilen benzetim sonuçlarına göre VNPER protokolü paket teslim etme oranı, uçtan uca ortalama gecikme ve toplam işlem hacmi metriklerine göre en başarılı protokol olmuştur. İletilen veri miktarı veya araç sayısının az olduğu

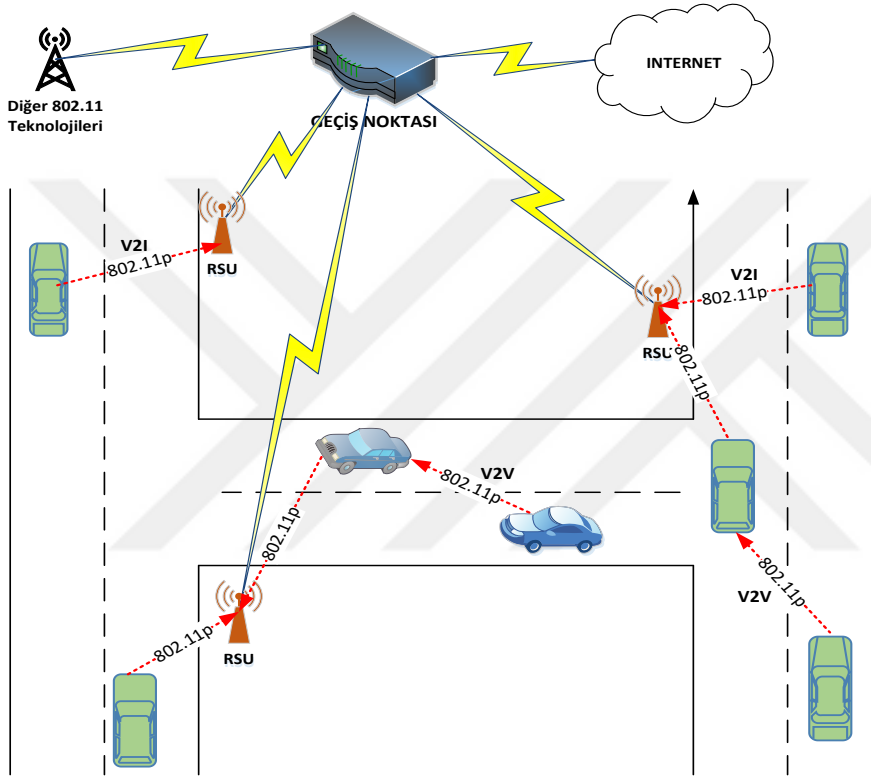
ortamlarda VNPER protokolü diğer protokollere göre daha masraflı olmasına rağmen, iletilen veri miktarı veya araç sayısı arttıkça VNPER protokolü diğer protokollere göre daha az masraflı duruma geçmektedir. VNPER protokolünün yönlendirme masrafındaki artış oranı, diğer protokollere göre daha azdır. Bundan dolayı, ağdaki araç sayısı veya veri miktarındaki artışın getirmiş olduğu sorunlara karşı en dayanıklı protokol VNPER olmuştur. VNPER protokolünün ağdaki büyümelere ve artışlara karşı dayanıklı olması, diğer protokollere oranla daha ölçeklenebilir bir yapıya sahip olduğu sonucuna ulaştırmaktadır.

Bu tezin ikinci bölümünde araçsal ağların mimarisi ve temel özelliklerinin yanı sıra literatür taramasına yer verilmiştir. Üçüncü bölümde VNPER protokolünün yapısı, algoritması ve çalışma mimarisi anlatılmıştır. Dördüncü bölümde performans ölçümü için kullanılan metriklerden, uygulama ortamı üzerinde kullanılan model ve senaryolardan alınan sonuçlara yer verilmiştir. Beşinci bölümde, bu tez kapsamında yapılan çalışmadan elde edilen sonuçlar özetlenmiştir ve gelecekte yapılması planlanan çalışmalardan bahsedilmiştir.



2. ARAÇSAL TASARSIZ AĞ MİMARİSİ

Araçsal ağ, kentsel alanlarda ve otoyollarda mevcut bir ağ alt yapısına ihtiyaç duymadan araçların birbirleri ile kablosuz bağlantı üzerinden oluşturdukları yeni nesil bir ağ teknolojisidir. Hareketli tasarsız ağlarının bir türevi olarak ortaya çıkartılmıştır. Gelecekte akıllı şehirlerin oluşturulması sürecinde araçsal ağların kullanılmasına kesin gözüyle bakılmaktadır. Şekil 2.1’de VANET’lerin temel mimarisi gösterilmiştir.



Şekil 2.1. Araçsal tasarsız ağ mimarisi

Araçsal ağlar bağlantı şekillerine göre, araçtan araca ve araçtan alt yapıya olarak iki sınıfa ayrılmaktadır. Literatürde araçsal ağlara yönelik çalışmalar, daha önce de bahsedildiği gibi iki sınıfa ayrılmıştır. Her sınıf başlı başına ele alınmıştır ve ilgili sınıfa özgü çalışmalar yapılmıştır.

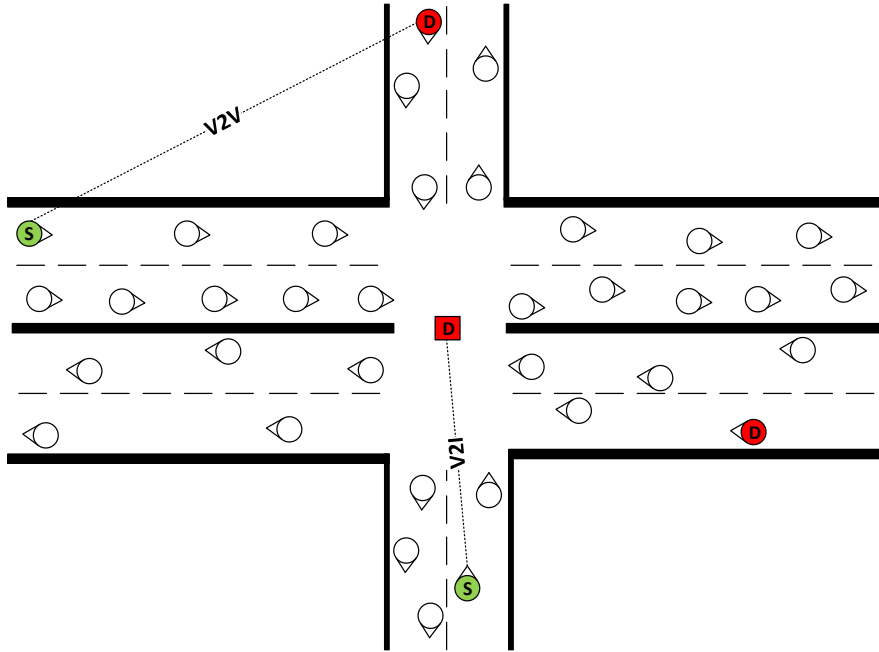
2.1. Araçtan Altyapıya Bağlantı (V2I)

Araçların sabit bir alt yapıyla oluşturdukları bağlantı şekli, araçtan alt yapıya (V2I) olarak ifade edilmektedir. V2I iletişimi VANET’lerin sabit alt yapı kısmını oluşturmaktadır. Önceden belirlenmiş kavşak gibi belirli noktalara kablosuz yönlendiriciler yerleştirilir. Literatürde bu kablosuz yönlendiriciler yol kenar birimi (RSU) olarak adlandırılmıştır.

Hareket halindeki araçların yol kenar birimleri ile oluşturdukları ağ türü, araçtan alt yapıya bağlantı olarak ifade edilir. RSU'ların internet gibi diğer ağ türleri ile bağlantısı mevcuttur. Araçlar bu noktalara geldikleri zaman diğer ağ türlerine veri gönderip alabilirler. Bu sayede, farklı ağ türleri ile araçsal ağların birbirleriyle bağlantısı sağlanmış olmaktadır. Özetle, araçların yol kenarlarındaki bazı noktalarda bulunan RSU olarak adlandırılan kablosuz yönlendiriciler ile kurduğu iletişimidir.

2.2. Araçtan Araca Bağlantı (V2V)

Otoyollar üzerindeki araçların birbirleriyle aralarında oluşturdukları bağlantıyı ifade etmektedir. Bu kısım, hızlı hareket eden MANET ağlar olarak düşünülebilir. Halihazırda sabit bulunan hiçbir ağ elemanı bulunmamaktadır. Tamamen araçların kendisi arasında oluşturdukları bağlantılardır. Araçsal ağ denilince aslında araçların birbirleri arasındaki iletişimi akla gelmektedir. Literatürde araçsal ağlara yönelik yapılan çalışmaların birçoğu araçtan araca bağlantı üzerine yoğunlaşmıştır. Araçtan araca bağlantı literatürde V2V olarak adlandırılmaktadır. Şekil 2.2'de VANET'lerdeki bağlantı modelleri gösterilmiştir. Şekil 2.2'de görüldüğü gibi, doğrudan iki düğüm arasındaki iletişim V2V olarak adlandırılırken, düğüm ile kavşak noktasında bulunan yol kenar birimi arasındaki iletişim ise V2I olarak adlandırılmıştır.



Şekil 2.2. VANET iletişim modelleri

2.3. Araçsal Ağların Karakteristik Özellikleri

VANET'leri diğer yeni nesil ağ türlerinden ayıran kendisine özgü birçok karakteristik özelliği bulunmaktadır. MANET'lerin bir türevi olarak kabul edilmesine rağmen, MANET'ler için geliştirilen protokollerin VANET ortamında kullanımı uygun değildir. Araçsal ağlara özgü birçok özellikten dolayı verimli sonuçlar elde edilememiştir. VANET ortamında düğümlerin hızları MANET'lere oranla çok daha fazladır. Ayrıca, belirli bölgelerde düğümler öbekleşerek oldukça yoğun bir ağ ortamı oluşturabilmektedir. Otoyol gibi hareket yönünün ve ivmenin daha az olduğu ortamlarda ise araç yoğunluğu oldukça seyrek olabilmektedir. Bunun gibi özelliklerden dolayı araçsal ağların yapısına özgü protokollere ihtiyaç duyulmaktadır. Araçsal ağlara özgü karakteristik özellikler başlıca; yüksek hız, hareketlilik modeli ve ağ yoğunluğu olarak 3 kısımda değerlendirilebilir.

2.3.1. Yüksek hız

V2V tabanlı VANET ortamında paketlerin hedeflerine doğru yönlendirme işleminin gerçekleştirilmesini etkileyen en önemli faktör, düğümlerin yüksek hızlara sahip olmasıdır. Yüksek hızlardaki düğümlerin hareketlerinden dolayı topoloji anlık olarak hızlı bir şekilde değişmektedir. Araçların hızları 200 km/saate kadar çıkabilmektedir. Yüksek hız, araçlar arasındaki iletişim süresinin kısılmasına ve ağ topolojisinin sürekli olarak hızlı bir şekilde değişmesine neden olmaktadır. Özellikle karşı yönlerden birbirlerine doğru yaklaşan iki araç arasındaki iletişim süresi oldukça kısa olmaktadır. Araçların hızları ile iletişim süresi ters orantılı olarak değişmektedir. Araç hızları arttıkça iletişim süresi kısalmaktadır. Bunun sonucunda araçlar arasında bağlantı kopmaları yaşanmaktadır. Araçların iletişimleri sırasında bağlantı süresini artırmak için, sinyal gücünü artırmak bir çözümdür. Ancak bu durum şehiriçi gibi trafiğin yoğun olduğu ortamlarda ağın verimliliğinin düşmesine neden olmaktadır. Bant genişliğinin etkin kullanımı azalacağından dolayı veri bağı katmanında olası çakışmalar meydana gelmektedir. Bunun sonucunda ağın performansı olumsuz yönde etkilenmektedir.

Yönlendirme algoritmasının yeterli bir performansa sahip olması için araç hızları, yönü ve ivmesi gibi bir çok bilgiye ihtiyaç duyulmaktadır. Bu bilgiler VANET ortamındaki araçlardan toplanmaktadır. Ancak, araçların yüksek hızlarda olması bu bilgilerin toplanabilmesini zorlaştırmaktadır. Araç hızlarının ve hareketlerinin şehiriçi gibi ortamlarda

yavaş olması ise farklı problemlerin ortaya çıkmasına neden olmaktadır. Araç hızlarının düşük olması, ilerleme hızını yavaşlatmaktadır. Bunun sonucu olarak da birim alana düşen araç sayısını artırmaktadır. Aşırı yoğunluğu nedeniyle ortamda çok fazla paket dolaşmaktadır. Bundan dolayı, ağda tıkanıklık ve paket kayıpları meydana gelmektedir.

2.3.2. Hareketlilik modeli

Araçtan araca VANET'leri oluşturan düğümlerde hareketlilik oldukça fazladır ve hızlı bir şekilde gerçekleşmektedir. Ancak, düğümlerin hareketleri tamamen rastlantısal değildir. Mevcut yolları kullanmak zorunda oldukları için, hareketleri bu yolların sınırları içerisinde rastlantısaldır. Düğümlerin sınırlı rastlantısal hareket edebilmesi durumu, araçların bir sonraki hareketlerinin tahmin edilebilmesine imkan tanıyabilir. Son zamanlarda, hareketlilik modellerini dikkate alan çalışmalar yapılmaya başlanmıştır [5-7].

Hareketlilik modelini dikkate alan yönlendirme protokolleri, diğerlerine göre daha başarılı sonuçlar elde etmektedir. Araçların hareketlerini belirleyen yollar 3 sınıfa ayrılabilir. Aşağıda belirtilen farklı yol türleri farklı hareket modellerini ortaya çıkarmaktadır.

Şehiriçi : Araç yoğunluğun en fazla olduğu ortamdır. Birçok kavşak ve alternatif yolların bulunmasından dolayı araçların rotalarını tahmin etmek oldukça zordur. Araç hızları diğer ortamlara göre daha düşüktür. Yolların etrafındaki yapılardan dolayı kapsama alanı ve bağlantı kopmaları yaşanması muhtemeldir.

Kırsal yollar : Araç yoğunluğu ve hızı orta seviyededir. Kavşak ve alternatif yolların miktarı şehiriçi ortamlara göre daha azdır. Bundan dolayı, hareket tahmini yapılabilmesi şehiriçi ortamlara göre daha kolaydır.

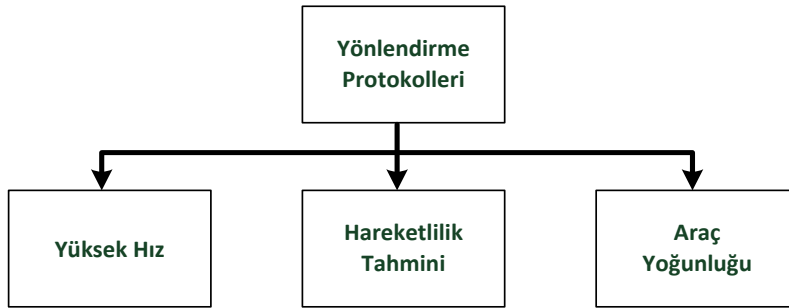
Otoyollar : Araç diziliminin en düzgün olduğu ortamdır. Araçların hızları, ilerleme yönü ve ivme değişimi diğer ortamlara göre oldukça azdır. Araç yoğunluğu oldukça seyrek. Araç hızının en yüksek olduğu ortamdır. Hareketlilik tahmini diğer ortamlara göre daha kolay yapılabilir.

2.3.3. Ağ yoğunluğu

V2V tabanlı araçsal ağlarda düğümler arasındaki bağlantı durumu değişimi, MANET'lere oranla çok daha fazla olmaktadır. Ağın seyrek olması aynı mesajın aynı araç tarafından fazla sayıda yayınlanmasına neden olmaktadır. Araç yoğunluğunun fazla olduğu ortamlarda ise bu durum tam aksine işlemektedir. İlgili mesaj önceden belirlenmiş belli araçlar tarafından yayınlanmalıdır. Her aracın doğrudan iletişime dahil olması, iletişim kanalında yoğunluğa yol açmaktadır. Bunun sonucunda ortama erişim problemi ortaya çıkmaktadır. Son zamanlarda araç yoğunluğunu dikkate alan çalışmalara literatürde rastlanmaktadır [8-10].

2.4. Araçsal Ağ Yönlendirme Protokolleri

Ağdaki verilerinin verimli bir şekilde kaynaktan hedefe gönderilebilmesi yönlendirme protokolünün yeteneklerine bağlıdır. VANET'ler için geliştirilen farklı yönlendirme protokollerinin bir gereksinimi karşılayabilmek için bazı özelliklerden ödün vermesi gerekebilmektedir. Araçtan araca iletişim için tasarlanan VANET yönlendirme protokolleri bu tez kapsamında hız farkındalığı, hareketlilik tahmini ve araç yoğunluğu özelliklerine göre Şekil 2.3'de gösterildiği gibi sınıflandırılmıştır.



Şekil 2.3. Araçtan araca yönlendirme protokollerinin sınıflandırılması

VANET'lerin yaygın şekilde kullanılmaya başlanmasıyla birlikte, video ve ses gönderiminin araçsal ağlarda da kullanımının oldukça yaygınlaşacağı tahmin edilmektedir. Video ve ses verilerine normal veri gibi yaklaşılması performans açısından uygun olmayacaktır. VANET'lerde video ve ses verilerinin etkin şekilde taşınabilmesinin birçok avantajı bulunmaktadır. Örneğin, trafik kazasının anında video ve ses verilerinin canlı olarak diğer araçlara ve kurtarma merkezine gönderilmesi önemli faydalar sağlar. VANET'lerde multimedia verilerinin taşınması sırasında kullanılan performans ölçüm metriklerinin

mevcut performans ölçüm metriklerinden farklı olması gerekmektedir. Mevcut QoS'e ait olan paket gecikmesi ve paket kaybı gibi metriklerin kullanılmasının doğru bir ölçüm olmayacağı ileri sürülmektedir. Bu yüzden sinyal-gürültü oranı gibi farklı QoE metriklerinin kullanılması gerekmektedir. Proaktif yönlendirme algoritmalarının multimedia verisinin taşınması için uygun olmadığı, reaktif yönlendirme yaklaşımlarının daha başarılı olduğu literatürde ifade edilmektedir [11].

Literatür incelendiğinde yakın zamanda yapılan çalışmalarda yüksek hız odaklı çalışmaların yanı sıra, hareketlilik şekli ve trafik yoğunluk farkındalığı üzerinde de durulduğu gözlemlenmiştir. Araçların hareket şekillerine göre bir sonraki zaman diliminde nerede olacakları önceden tahmin edilerek verimlilik artırılmaya çalışılmıştır. Araç yoğunluğu odaklı yapılan çalışmalarda ise, araç yoğunluğunun ortaya çıkardığı sorunlar en aza indirilmeye çalışılmıştır. Bu tez kapsamında, literatürde bulunan VANET yönlendirme protokolleri hız farkındalığı, hareketlilik tahmini ve araç yoğunluğu dikkate alınarak sınıflandırılmıştır.

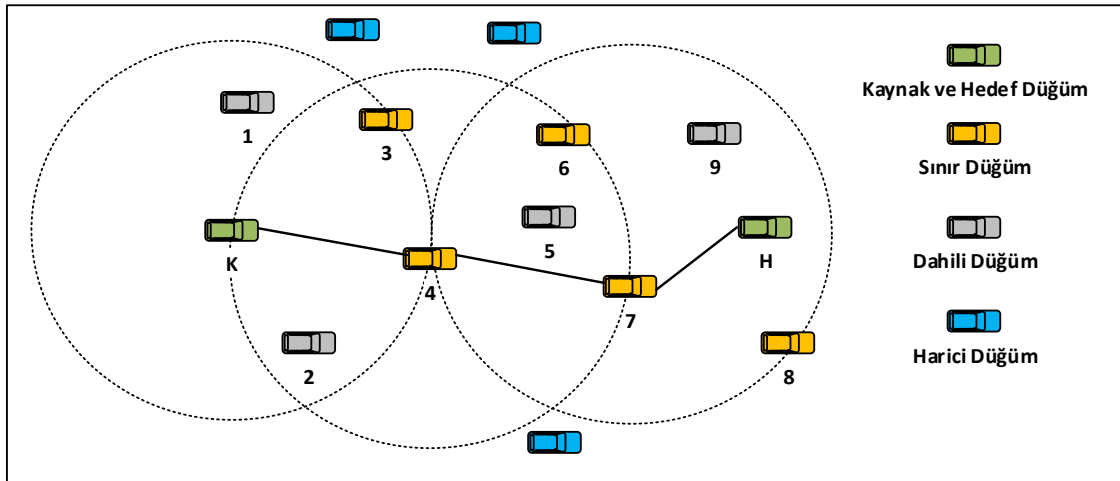
2.4.1. Hız farkındalıklı V2V yönlendirme protokolleri

Karp ve Kung 2000 yılında yaptıkları çalışmada, GPSR (Greedy Perimeter Stateless Routing - Anında Tepkimeyi Çevre Bağımsız Yönlendirme) isimli kablosuz ağ yönlendirme protokolü geliştirmişlerdir [12]. Mobil ortamlar için üretilmiş gecikme toleransı olmayan pozisyon tabanlı bir yönlendirme protokolüdür. GPSR kentsel alanlarda kullanılması uygun değildir. Alsaqour ve diğerleri 2012 yılında yaptıkları çalışmada GPSR protokolünün, araçların nispeten düzenli şekilde dağıldığı otoban gibi ortamlarda daha başarılı sonuçlar elde ettiği sonucuna ulaşmışlardır [13]. Komşu düğümlerin lokasyon bilgisi beacon mesajları (işaret mesajları) gönderilerek bildirilir. Protokolün yönlendirme algoritması greedy (aç gözlü) ve perimeter (çevre) olarak adlandırılan iki moddan oluşmaktadır. Greedy modda düğümler hedefe veri gönderecekleri sırada komşuları arasında hedefe en yakın olanı seçer ve anında gönderir. Verinin sıçratılacağı komşu düğüm seçimi sırasında başka hiçbir parametre dikkate alınmaz. Kaynak düğümün veriyi göndereceği hedefe kendisinden daha yakın bir komşu düğümü yoksa, perimeter mod denilen moda geçilir. Bu durumda düğüm veriyi doğrudan hedefe gönderir [12,13].

Naumov ve diğ erleri 2016 yılında yaptıkları çalışmada, GPSR+AGF (Advanced Greedy Forwarding - Gelişmiş Anında Tepkimeli Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [14]. GPSR+AGF protokolü, GPSR'deki sıklıkla topoloji değişimi sonucu ortaya çıkan düşük performansı iyileştirmek amacıyla geliştirilmiştir. GPSR protokolü ile ilgili iki tane sorun bulunmaktadır. Yüksek hızlarda hareketlerden dolayı, bir düğümün komşuluk tablosunda bulunan kayıtların çoğunun geçersiz durumda kalmasıdır. Beacon mesajlarının yayılım sıklığı artırılarak bu sorun çözülebilir. Ancak bu çözüm ağda çok fazla beacon mesajının dolaşmasına neden olacağından paket çakışmalarına neden olmaktadır. Bir diğ er sorun ise, yönlendirme paketlerinde hedef düğümün lokasyon bilgisi güncellenmediği için, verinin uygun hedefe gönderilememesidir. Bu sorunları çözmek için, Naumov ve diğ erleri beacon paketleri içerisine düğümlerin yön ve hız bilgilerini eklemiştir. Alınan beacon mesajlarındaki yön ve hız bilgilerine göre düğümler komşuluk tablosunda güncelleme yapmaktadır. Paket teslim etme oranını, GPSR protokolüne göre 3 katına çıkarmışlardır.

Naumov ve Gross 2007 yılında yaptıkları çalışmada, CAR (Connectivity Aware Routing - Bağlantı Farkındalıklı Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [15]. CAR protokolü hedef düğümün lokasyonunda bir değişim olduğunu tespit edebilir. Hello paketlerini yayarak komşu düğümlerin hızlarını ve yönlerini öğrenir. Hello paketlerini alan düğümler paketi gönderen komşu düğümüne ait hız ve yön bilgisini komşuluk tablosunda saklar. Fazla sayıda aracın olduğu ortamlarda kaynak ve hedef düğüm arasında oldukça fazla rota seçeneği oluşmaktadır. Bu rotalar arasından en uygun olanı seçerek, yönlendirme tablosunda saklamaktadır. Beacon mesajları veri paketlerinin sonuna eklenir. Beacon mesajlarının sıklığı komşu düğümlerin sayısına göre değişir. Yüksek hızdan dolayı oluşan rota bozulmalarını bekçi olarak adlandırdıkları mekanizma ile azaltmışlardır. Bekçi mekanizması iki moddan oluşmaktadır. Uç noktalardaki düğümlerin pozisyonu ve yönü değişirse durağan bekçi modu aktif hale getirilerek paket kaybı olmadan rota değişimi gerçekleştirilir. Uç noktadaki düğümlerin yönü iletişim yönüne ters olarak değişirse, hareketli bekçi modu aktif hale getirilir. Hareketli bekçi moduna geçen aradaki düğüm, kendisi hedef düğüm gibi davranır. Gelen paketleri kendisinde toplayarak gerçek hedef düğümün yeni yön ve hareket bilgisine göre rota keşfi yapar ve paketleri asıl hedefe yönlendirir. Kaynak düğümüne ise kendi modu hakkında bilgi mesajı gönderir.

Raw ve Lobiya 2010 yılında yaptıkları çalışmada, BMFR (Border node-based Most Forward within Radius - Kapsama Alanı Sınır Düğüm tabanlı Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [16]. BMFR protokolü kaynak ve hedef düğüm arasındaki sıçrama sayısını en aza indirmek için paketi gönderen düğümün kapsama alanındaki sınır düğümü kullanır. Düğümler dahili, harici ve sınır olmak üzere üç sınıfa ayrılır. Bir düğümün kapsama alanı içerisindeki tüm düğümler dahili düğümler olarak kabul edilirken, kapsama alanı dışında olan düğümler ise harici düğüm olarak kabul edilir. Kapsama alanının sınırında olan düğümler ise sınır düğüm olarak adlandırılır. Kapsama alanı içerisindeki düğümler hakkında bilgiyi beacon mesajlarının değişimi ile öğrenir. Dahili düğümlerin hepsi bir sıçrama uzaklığında olan düğümlerdir. Kapsama alanı içerisindeki tüm düğümler hakkında bilgi edinildikten sonra, sıçrama noktası olarak sınır düğüm seçilir. Çünkü, hedefe en yakın olan düğüm sınır düğüm olarak kabul edilir. Şekil 2.4’de K düğümü H düğümü ile iletişim kuracağı zaman, sınır düğümleri olan 3 ve 4 nolu düğümler arasından hedefe en yakın olan düğümü seçer. Bu durumda 4 nolu düğüm sıçrama noktası olarak seçilir. 4 nolu düğüm de benzer şekilde, sınır düğümleri olan 6 ve 7 nolu düğümler arasından 7 nolu düğümü sıçrama noktası olarak seçer. 7 nolu düğüm hedefe doğrudan ulaşabilmektedir.



Şekil 2.4. BMFR protokolünün sıçrama noktası seçimi

Song ve diğerleri 2010 yılında yaptıkları çalışmada, CBDRP (Cluster Based Directional Routing Protocol - Kümeleme Tabanlı Yönlü Yönlendirme Protokolü) isimli kümeleme yaklaşımını temel alan araçsal ağ yönlendirme protokolü geliştirmişlerdir [17]. Sadece aynı yöne giden araçlar kümeye ayrılır. VANET ortamı için geliştirilen 802.11p standartına göre,

bir aracın kapsama alanı, yaklaşık olarak 1000 metre yarıçapındadır. CBDRP protokolünde küme sınırı olarak yarıçapın yarısı alınmıştır. Bu alan içerisindeki araçlar arasından bir tanesi küme başı olarak seçilir. Küme elemanları başka bir küme içerisinde bulunan bir araç ile iletişime geçeceği zaman, sıçrama noktası olarak küme başı olan aracı seçmektedir. Kaynak düğümün küme başı aracı tarafından hedef düğümün bulunduğu küme başına doğru rota keşfi yapılır. Paketler küme başı araçlar üzerinden hedef düğüme ulaştırılır. Küme başı düğümün seçilmesi CBR protokolündekine benzerdir. CBR protokolünün aksine, araçların hız ve yön bilgisi de göz önünde bulundurulur.

Araçsal ağlarda iletişim türü olarak tek yönlü yayın, çoklu yayın ve her yöne yayın modelleri kullanılmaktadır. VANET ortamında sadece iki aracın birbirleri ile iletişimi tek yönlü yayın olarak ifade edilir. Bir kaza anında bu bilginin sadece ilgili alanda bulunan araçlara gönderilmesi çoklu yayına örnektir. Kaza bilgisinin sadece kazanın olduğu alanda bulunan araçlara gönderilmesi, VANET ortamında gereksiz paket dolaşımını azaltacaktır. Araçların rota keşfi yapmak, küme başı düğümü bulmak gibi durumlarda kullandığı yayın türü ise her yöne yayına örnektir. Her yöne yayında VANET ortamındaki tüm araçlara mesaj gönderilir. Geliştirilen VANET yönlendirme protokollerinin bir kısmı çoklu yayın yaklaşımına göre geliştirilmiştir. Literatürde bu protokoller geocast tabanlı yönlendirme protokolleri olarak adlandırılmıştır.

Bachir ve Benslimane 2003 yılında yaptıkları çalışmada IVG (Inter Vehicle Geocast - Araçlar Arası Coğrafi Yayın) isimli çoklu yayın tabanlı araçsal ağ yönlendirme protokolü geliştirmişlerdir [18]. IVG protokolü bir aracın trafik kazası veya sel, deprem gibi doğal afetlerin diğer araçlara aktarılması için tasarlanmış bir protokoldür. GPS tabanlı bir yönlendirme protokolüdür. IVG protokolünde aracın yönüne göre bir risk alanı oluşturulur. Herhangi bir araçtan gelen mesajın gerçekten diğer araçlara gönderilip gönderilmeyeceğinin tespit edilmesi için mesajlar belli bir süre bellekte bekletilir. Bu süre içerisinde benzer mesajların diğer araçlardan da gelmesi beklenir. Aynı mesajın belli bir süre içerisinde bir çok araçtan gelmesi durumunda hedefe doğru mesaj gönderilir. IVG protokolü, hareketliliğin fazla olduğu ortamlarda oldukça fazla yönlendirme yükü ile çalışmaktadır.

Maihofer ve Eberhardt 2004 yılında yaptıkları çalışmada CGR (Cached Geocast Routing - Önbellekli Coğrafi Yayın tabanlı Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [19]. CGR protokolü lokal minimum probleminin üstesinden gelebilmek

için diğer düğümlerden gelen paketleri anında hedefe doğru göndermez. Lokal minimumun problemi ile karşılaşıldığında, yönlendirme katmanında paketleri tampon bir bellekte saklar. Yeni bir komşu düğüm bulununca ya da mevcut komşu düğümlerde bir pozisyon değişikliği olduğunda bellekteki paketler gönderilir. CGR protokolünün yönlendirme yükünü ve gecikme miktarını azalttığı görülmüştür.

Kihl ve diğerleri 2007 yılında yaptıkları çalışmada ROVER (Robust Vehicular Routing - Sağlam Bağlantılı Araçsal Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [20]. ROVER protokolü çoklu yayın yapan bir yönlendirme protokolüdür. VANET ortamında sadece kontrol paketleri ağa yayın mesajları olarak gönderilir. Veri paketleri ise tek yönlü yayın şeklinde gönderilir. Belirli bir alan içerisinde bulunan tüm araçlara veri göndermek için tasarlanmış bir protokoldür. Bir kaza anında etrafta bulunan araçların kaza bilgisine sahip olması gerekir. Kaza anında etrafta bulunan araçların oluşturduğu alan literatürde ZOR [21] (Zone of Relevance) olarak adlandırılmıştır. Bir araca mesaj geldiği zaman, mesajın ZOR içerisindeki bir araçtan gelip gelmediğine bakılır. Kendi ZOR alanı içerisinde ise mesaj alınır. ZOR alanı içerisinde rota keşfi reaktif olarak yapılır. ROVER protokolü VANET ortamında dolaşan gereksiz bir çok paket üretmektedir. Bundan dolayı, ağda tıkanıklık oluşmasına ve gecikme sürelerinin artmasına neden olmaktadır.

Lee ve diğerleri 2002 yılında yaptıkları çalışmada D-ODMRP (Destination driven On Demand Multicast Routing Protocol - Hedef Güdümlü İstek Tabanlı Çoklu Yayın Yönlendirme Protokolü) isimli çoklu yayın tabanlı araçsal ağ yönlendirme protokolü geliştirmişlerdir [22]. D-ODMRP protokolü etkin bir şekilde çoklu yayın yapabilmek için geliştirilen bir yönlendirme protokolüdür. VANET ortamını çoklu yayın gruplarına ayırır. Hedefin içinde bulunduğu çoklu yayın grubuna ulaşmak için en az sıçrama sayısı metriğini kullanır. Çoklu yayın gruplarını belirlemek için keşif istekleri yayın mesajı olarak grup içerisine gönderilir. Yayın mesajları çoklu yayın grubu dışarısına çıkartılmaz. Bundan dolayı ağda tıkanıklık ve gereksiz paket gönderimine neden olmamaktadır.

Durresi ve diğerleri 2005 yılında yaptıkları çalışmada BROADCASTCOMM isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [23]. BROADCASTCOMM protokolü otoyollardaki gibi araçların hiyerarşik bir yapı oluşturdukları VANET ortamları için tasarlanmış bir protokoldür. Otoyol, uzunlukları birbirine eşit olan sanal hücrelere bölünür. Otoyol üzerindeki araçlar iki seviyeye ayrılır. Bir hücre içerisinde bulunan tüm araçlar birinci

seviyededir. Hücrenin merkezine yakın olan düğümler ise ikinci seviyededir. Belirli bir zaman aralığında ikinci seviyedeki düğümler küme başı gibi hareket etmektedir. Hücre içerisindeki diğer araçlardan gönderilen mesajları yönetmektedir. Her yöne yayın tabanlı bir protokol gibi çalışmaktadır.

Maia ve diğerleri 2012 yılında yaptıkları çalışmada HyDi (Hybrid Data Dissemination - Melez Veri Yayınımlama) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [24]. HyDi otoyollarda kullanım için tasarlanmıştır. Araçların sürekli veya kısmi bağlantıya sahip olduğu VANET ortamları için geliştirilmiş bir yönlendirme protokolüdür. Yayın mesajlarını belirli bir yöne doğru yönlendirerek hedef aracı bulmaya çalışmaktadır. Daha az yayın mesajı kullanarak rota tespiti yapmaktadır. Gönderici ve alıcı araç tabanlı iki farklı yaklaşıma sahiptir. Gönderici tabanlı yaklaşımda, mesajlar öndeki araçlardan alınırken, alıcı tabanlı yaklaşımda mesajlar arkadaki araçlardan alınır. Mesajın gönderimi sırasında herhangi bir araç ile iletişim kurulamazsa, veriler bellekte saklanır ve seçilen yaklaşıma göre bağlantı sağlanan araca veriler gönderilir. Trafiğin yoğun olduğu ortamlarda az miktarda gecikme ile paket teslim edilmektedir. Ayrıca, daha az sayıda sıçrama yapılarak hedefe ulaşılmaktadır.

Ahmed ve Kanere 2006 yılında yaptıkları çalışmada SKVR (Scalable Knowledge based Vehicular Routing - Ölçeklenebilir Bilgi tabanlı Araçsal Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [25]. SKVR protokolü VANET ortamını, düğümlerin sahip olduğu rota bilgisine göre alan içi ve alanlar arası olarak iki kısma ayırır. Aynı rota bilgisine sahip olan düğümler alan içi düğüm olarak adlandırılırken, farklı rota bilgisine sahip olan düğümler alanlar arası düğüm olarak adlandırılır. Alanlar arası düğümlerde iletişim olacağı zaman, öncelikle hedefin ait olduğu alan tespit edilir. Hedef alana paketler iletildikten sonra, ilgili alan içerisindeki düğümler tarafından hedefe paketler teslim edilir. Araçlar arası seyrek bağlantının olduğu VANET ortamları için tasarlanmıştır.

Zhao ve Cao 2008 yılında yaptıkları çalışmada VADD (Vehicle Assisted Data Delivery - Araç Destekli Veri Gönderme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [26]. VADD protokolü araçlar arasında bağlantıların seyrek olduğu ve bağlantı kopmalarının çok sık olduğu VANET ortamları için tasarlanmış bir yönlendirme protokolüdür. Yönlendirme işlemi kavşak noktalarındaki düğümler tarafından yapılmaktadır. Diğer düğümler gelen paketleri kavşak noktasına doğru iletmekten sorumludur. Kavşak, düz yol

ve hedef olarak adlandırılan 3 farklı paket modu en uygun rotayı seçmek için kullanılmaktadır. Kavşak noktasından uzakta olan paketler düz yol modundadır. Kavşak noktasındaki aracın kapsama alanına girince paketin modu değiştirilerek kavşak moduna alınır. Hedef araca ulaşan paketler ise hedef moduna çevrilir. Her bir paket modunda farklı yönlendirme işlemleri kullanılmaktadır. Yüksek hızdan dolayı ortaya çıkabilecek sorunları çözmeye yönelik geliştirilen yönlendirme protokolleri Çizelge 2.1’de farklı özelliklere göre karşılaştırmalı olarak sunulmuştur.

Çizelge 2.1. Yüksek hız odaklı V2V yönlendirme protokollerinin karşılaştırılması

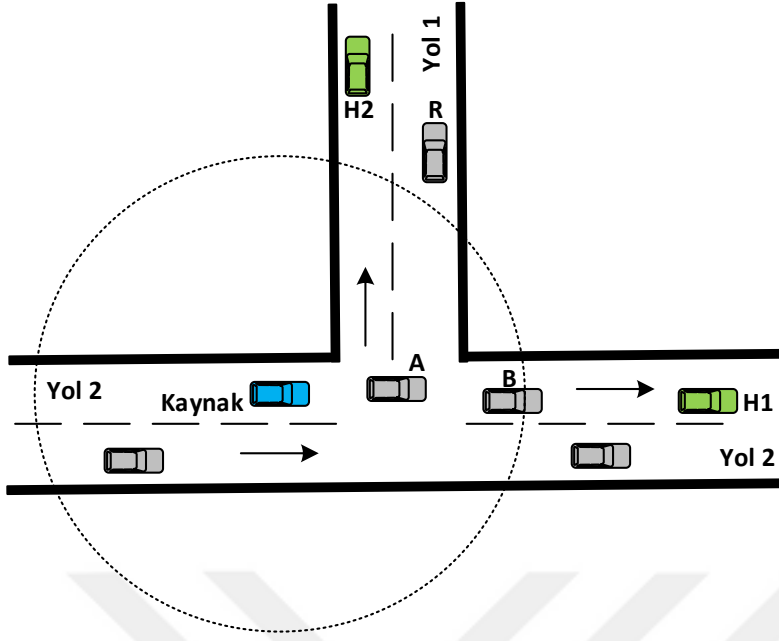
Protokoller	Yönlendirme Algoritması	Harita Kullanımı	GPS Kullanımı	Araç Yoğunluğu	Taşı-Gönder Desteği	İletişim Ortamı
GPSR	Greedy	Yok	Var	Seyrek	Yok	Otoyol
GPSR+AGF	Greedy	Yok	Var	Orta	Yok	Kentsel
CAR	Greedy	Var	Var	Orta	Yok	Kentsel
B-MFR	Greedy	Var	Var	Çok	Yok	Kentsel
CBDRP	Çoklu Sıçrama	Var	Yok	Orta	Var	Kentsel
IVG	Çoklu Sıçrama	Yok	Yok	Seyrek	Yok	Otoyol
CGR	Çoklu Sıçrama	Yok	Yok	Orta	Yok	Kentsel
ROVER	Çoklu Sıçrama	Yok	Yok	Çok	Yok	Kentsel
D-ODMRP	Çoklu Sıçrama	Yok	Yok	Seyrek	Var	Otoyol
BROADCOMM	Çoklu Sıçrama	Yok	Yok	Orta	Var	Otoyol
HyDi	Çoklu Sıçrama	Yok	Yok	Seyrek	Var	Otoyol
SKVR	Greedy	Yok	Yok	Orta	Var	Kentsel

2.4.2. Hareketlilik tahmini odaklı V2V yönlendirme protokolleri

Mo ve diğerleri 2006 yılında yaptıkları çalışmada MURU (Multihop Routing Protocol - Çoklu Sıçrama tabanlı Yönlendirme Protokolü) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [27]. MURU protokolü aracın pozisyonu, hızı ve yönüne göre rotaların uygunluğunu tahmin etmektedir. Elde ettiği rota kalitesine göre de rota seçimini yapmaktadır. Hız, yön ve araç pozisyonunu dikkate alarak EDD olarak adlandırılan yeni bir metrik öne sürmüşlerdir. Hedef düğümün lokasyon bilgisi ve cadde harita bilgisine göre, kaynak düğüm tarafından hedefe doğru giden rotalar arasından en kısa yol hesaplanır. En kısa yol bilgisi rota istek paketlerine cevap olarak gönderilen paketlerin içerisine eklenir. İki düğüm arasındaki EDD metriği rota istek paketini alan düğüm tarafından hesaplanır. MURU

protokolü rota istek paketlerinin yayılımından dolayı oluşabilecek ölçeklenebilirlik sorununu en aza indirmek için budama metodunu kullanır. Bu sayede, rota istek paketlerinin ilgisi olmayan düğümlere gitmesi engellenir. MURU protokolü verimliliği daha yüksek olan rotaların kullanılmasını sağlamaktadır. Kentsel alanlarda kullanım için geliştirilmiş bir protokoldür.

Lee ve diğerleri 2007 yılında yaptıkları çalışmada GpsrJ+ isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [28]. GpsrJ+ protokolü sezgisel tahmine dayalı bir algoritmaya sahiptir. Kavşak noktalarında duvar, ağaç gibi engellerin olmadığını varsayar. Hedef düğümü tespit etmek için iki sıçrama sayısı uzaklığa kadar gidebilen beacon mesajları kullanmaktadır. Her bir düğüm koordinat bilgilerini ve hangi yol üzerinde olduğu bilgisini beacon mesajlarına ekler ve komşularına gönderir. Bu bilgilere göre kavşak noktalarında hangi düğümün sıçrama noktası olarak seçileceğine karar verilir. Kaynak düğüm ile hedef düğüm farklı yöne doğru hareket ediyorsa kavşak noktasındaki düğüm sıçrama noktası olarak seçilir. Aynı yol üzerinde ve aynı yönlere doğru hareket ediyorsa, kavşak noktasındaki düğüm yerine daha uzaktaki düğüm seçilir. Şekil 2.5’de A düğümü kendi koordinat bilgisini ve yol 1, yol 2 üzerinde olduğu bilgisini komşularına bildirir. Benzer şekilde B düğümü, koordinat bilgisini ve yol 2 üzerinde olduğunu, R düğümü de koordinat bilgisini ve yol 1 üzerinde olduğu bilgisini komşularına bildirir. Bu bilgilere göre, kaynak düğüm H1 düğümü ile iletişime geçeceği zaman, B düğümü ile aynı yol üzerinde ve aynı yöne doğru hareket ettikleri için A düğümü yerine B düğümünü sıçrama noktası olarak seçmektedir. Kaynak düğüm H2 düğümü ile iletişime geçeceği zaman, kaynak ve H2 düğümleri farklı yollar üzerinde oldukları için kavşak noktasındaki A düğümü sıçrama noktası olarak seçilmektedir.

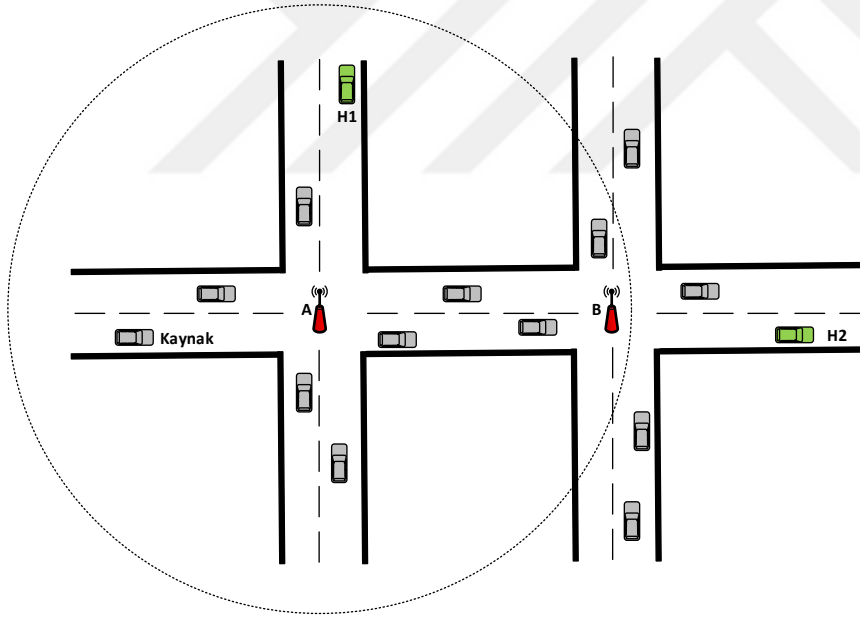


Şekil 2.5. GpsrJ+ protokolünün sıçrama noktası seçimi

Tonguz ve diğerleri 2007 yılında yaptıkları çalışmada DV-CAST (Distributed Vehicular Broadcast - Dağıtık Araçsal Yayın) isimli araçsal ağ yönlendirme protkolu geliştirmişlerdir [29]. DV-CAST protokolü paketlerin gereksiz ya da fazladan olup olmadığını tespit etmek için bayrak olarak adlandırılan bir değişken kullanır. Hello mesajlarını beacon olarak göndererek topoloji bilgisini rota keşfi için kullanır. DV-CAST protokolünde araçlar iyi bağlantıya sahip olanlar, zayıf bağlantıya sahip olanlar ve bağlantısı olmayan araçlar olarak 3 sınıfa ayrılmıştır. Zayıf bağlantıya sahip olan araçlar her yöne yayın mesajı alır almaz aynı yöne giden araçlara bu mesajı iletir. Bağlantısı olmayan araçlar ise almış olduğu verileri bellekte saklar. Herhangi bir araç ile bağlantı sağladığı zaman bellekteki verileri iletir. Belli bir süre içerisinde hiçbir araç ile bağlantı sağlayamaz ise verileri yok eder.

Nekovee ve Bogason 2007 yılında yaptıkları çalışmada EAEP (Edge Aware Epidemic Protocol - Engel Farkındalıklı Yaygın Protokol) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [30]. EAEP protokolü yüksek hareketliliğe sahip VANET ortamları için tasarlanmıştır. Bant genişliğini etkin bir şekilde kullanmayı amaçlamıştır. Araçlar arasında hello paketlerinin değiş tokuşunu en aza indirerek yönlendirme yükünü ve bant genişliği tüketimini azaltmaktadır. Tüm araçlar kendi pozisyon bilgisini beacon mesajları ile gönderirler. Bir araç önündeki ve arkasındaki komşu araçlar ile farklı tipte iletişim kurar. Alınan bir paketin arkasındaki araca veya önündeki araca gönderilip gönderilmeyeceğine karar verilir. Bu sayede, çoklu yayın miktarı azaltılmaktadır.

Korkmaz ve diğerleri 2004 yılında yaptıkları çalışmada UMB (Urban Multihop Broadcast - Kentsel Çoklu Sıçramalı Yayın) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [31]. UMB protokolü kentsel alanlarda karşılaşılan broadcast storm ve gizli düğüm problemlerini giderebilmek için tasarlanan her yöne yayın tabanlı bir yönlendirme protokolüdür. Şekil 2.6'da UMB protokolünün çoklu yayın mesajlarına yön vermesi gösterilmiştir. Kaynak ve H1 aracı, A yol birim ünitesinin kapsama alanında iken, H2 aracı B yol birim ünitesinin kapsama alanındadır. Kaynak araç, H1 ya da H2 ile iletişim kuracağı zaman, paketleri çoklu yayın ile komşu düğümleri üzerinden sıçratarak A yol kenar ünitesine gönderir. A yol kenar ünitesi H1 ile iletişim kuracağı zaman, paketleri sadece kuzey yönüne doğru çoklu yayın olarak gönderir. H2 ile iletişim kuracağı zaman ise sadece B yol kenar birimine doğru paketleri çoklu yayın ile gönderir. B yol kenar birimi ise sadece doğu yönüne doğru paketleri çoklu yayın ile iletir. Çoklu yayın miktarı azaltıldığı için, rota yükü ve ağ tıkanıklığı azaltılmıştır.



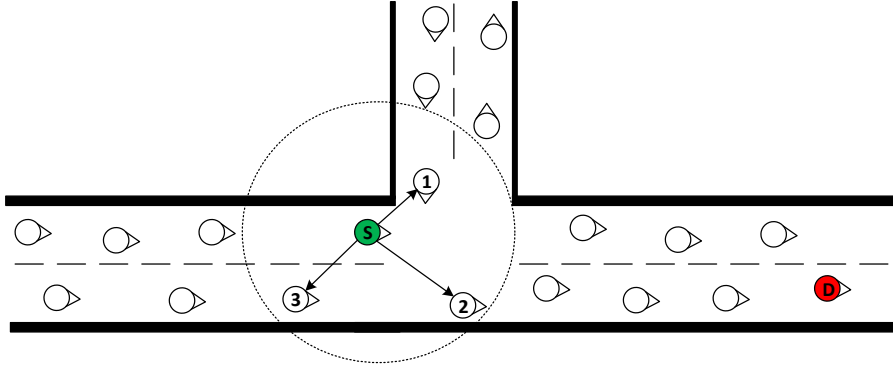
Şekil 2.6. UMB protokolünün her yöne yayın gönderimi

Luo ve diğerleri 2010 yılında yaptıkları çalışmada CBR (Cluster Based Routing - Kümeleme Tabanlı Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [32]. CBR protokolü araçsal ağlar için tasarlanmış kümeleme tabanlı bir yönlendirme protokolüdür. Bu protokol öncelikle VANET ortamını eşit karelere bölmektedir. Belli bir düğüm kümesi içerisinde bir düğüm küme başı olarak seçilir. Küme içerisinde veri gönderecek olan düğümler paketleri küme başı olan düğüme iletirler. Yönlendirme yükü diğer protokollere oranla daha azdır. Çünkü tüm düğümler hedefe yönelik bir rota keşif çalışması

yapmamaktadır. Küme başı düğüm, kendi kümesindeki düğümlere kendisine ait lokasyon ve koordinat bilgisini yayarak gönderir. Eğer ilgili alan içerisinde bir yol kenar birimi varsa küme başı olarak seçilir. CBR protokolünün en büyük dezavantajı araçlara ait hız ve yön bilgilerini dikkate almamasıdır.

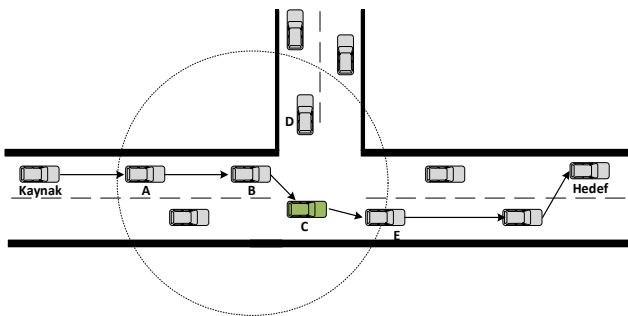
Lochert ve diğerleri 2003 yılında yaptıkları çalışmada GSR (Geographic Source Routing - Coğrafik Kaynak tabanlı Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [33]. GSR pozisyon tabanlı yönlendirme yaklaşımını, ağdaki topoloji bilgileri ile birleştiren bir yönlendirme protokolüdür. Cadde/sokak haritalarının kullanımı GSR protokolünün temelini oluşturur. Kentsel alanlarda kullanılmak amacıyla geliştirilmiştir. Hedef düğümün pozisyon bilgisini elde etmek için Camp ve arkadaşları tarafından geliştirilen reaktif lokasyon servisini (RLS) [34] kullanır. Hedef düğümün keşfedilmesi için hedef düğüm ile ilgili pozisyon isteği ağa gönderilir. Pozisyon isteğini alan düğüm hedef düğüme ait pozisyon bilgisine sahip ise, pozisyon cevap paketini isteği yapan düğüme gönderir. Paket teslim etme oranı AODV [35] ve DSR [36] protokollerine göre daha fazladır. GSR protokolü ağda ayrık düğüm kümelerinin olmadığını varsaymaktadır. Ayrıca, kontrol mesajı olarak hello paketlerini sürekli olarak ağa gönderdiği için, yüksek oranda yönlendirme yükü kullanarak işlem yapmaktadır.

Tian ve diğerleri 2003 yılında yaptıkları çalışmada SAR (Spatially Aware packet Routing - Uzaysal Boyut Farkındalıklı Paket Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [37]. SAR protokolü, GPSR ve GSR protokollerindeki lokal maksimum yaklaşımdan kaynaklanan verimsizliklerin üstesinden gelmek için geliştirilmiştir. Lokal maksimum durumu olduğu zaman paketi hedefine ulaştırmak için alternatif yollar bulmaktadır. GSR protokolü, Şekil 2.7'de S düğümü D düğümüne veri göndereceği zaman komşuları arasından kendisine en yakın komşusu olan 1 nolu düğümü sıçrama noktası olarak seçer. Ancak, hedef düğüm 2 nolu düğüm ile ulaşılabilir durumdadır. SAR protokolü ağın topolojisini önceden bildiği için sıçrama noktası olarak 2 nolu düğümü seçer. Bu sayede lokal maksimum probleminin üstesinden gelebilir.



Şekil 2.7. GSR ve SAR protokolü rota keşfi

Lochert ve diğerleri 2005 yılında yaptıkları çalışmada GPCR (Greedy Perimeter Coordinator Routing - Anında Tepkimeli Eş Güdümlü Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [38]. GPCR protokolü cadde/sokak haritaları kullanmaya ihtiyaç duymayan bir protokoldür. GPCR protokolü, sıçrama noktası düğümü olarak kavşak üzerindeki araçları seçmeye çalışır. Şekil 2.8’de protokolün sıçrama noktası düğüm seçimi gösterilmiştir. B düğümünün kapsama alanında olan C, D ve E düğümleri arasından C düğümünü sıçrama noktası olarak seçmektedir. Çünkü C düğümü kavşak noktasındadır ve kavşağı oluşturan yollardaki araçlar hakkında bilgi sahibidir. D düğümü kendisine en yakın komşu olmasına rağmen seçilmemiştir. Çünkü D düğümü üzerinden hedef düğüme ulaşamaz. Sıçrama noktası olarak E düğümünün seçilmesi de farklı bir topolojide yanlış bir seçime neden olabilir. Bu durumda lokal maksimum problemi ortaya çıkar. Paketler öncelikle C düğümüne gelir. C düğümü hedefe ait bilgiye sahip olduğu için doğru sıçrama noktasını belirler. GPCR protokolünde, C düğümü koordinatör olarak adlandırılır.



Şekil 2.8. GPCR protokolünün sıçrama noktası seçimi

Cha ve diğerleri 2012 yılında yaptıkları çalışmada GPGR (Grid-based Predictive Geographical Routing - Kare Bölgesel Alan tabanlı Öngörülü Coğrafik Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [39]. GPCR protokolü, kentsel alanlarda

kullanım için geliştirilmiş GPS tabanlı yönlendirme protokolüdür. VANET ortamının coğrafik haritası mantıksal olarak eşit boyutlarda karelere ayrılır. Ağa gönderilen toplu yayın paketleri en fazla 8 grid uzaklıktaki komşulara kadar gidebilir. Bu sayede ağın ölçeklenebilirliği artırılmıştır. GPCR protokolü araçlar arasındaki bağlantı kopma olasılığını ve lokal maksimum problemini grid mantığı kullanarak azaltmıştır. GPSR ve GPCR protokollerine oranla paket teslim etme oranı daha fazladır.

Hareketliliğin yoğun olmasından kaynaklı ortaya çıkabilecek sorunları çözmeye yönelik geliştirilen yönlendirme protokolleri Çizelge 2.2’de farklı özelliklere göre karşılaştırmalı olarak sunulmuştur.

Çizelge 2.2. Hareketlilik odaklı araçtan araca yönlendirme protokolleri

Protokoller	Yönlendirme Algoritması	Harita Kullanımı	GPS Kullanımı	Araç Yoğunluğu	Taşı-Gönder Desteği	İletişim Ortamı
MURU	Greedy	Var	Var	Orta	Yok	Kentsel
GpsrJ+	Greedy	Yok	Var	Çok	Yok	Kentsel
DV-CAST	Çoklu Sıçrama	Yok	Yok	Orta	Var	Otoyol
EAEP	Çoklu Sıçrama	Yok	Yok	Seyrek	Var	Otoyol
UMB	Çoklu Sıçrama	Yok	Yok	Seyrek	Var	Otoyol
CBR	Çoklu Sıçrama	Yok	Var	Orta	Yok	Kentsel
GSR	Greedy	Var	Var	Çok	Yok	Kentsel
SAR	Greedy	Var	Var	Orta	Yok	Kentsel
GPCR	Greedy	Var	Var	Çok	Yok	Kentsel
GPGR	Greedy	Var	Var	Orta	Yok	Kentsel

2.4.3. Ağ yoğunluğu odaklı yönlendirme protokolleri

Giudici ve Pagani 2005 yılında yaptıkları çalışmada STAR (Spatial and Traffic Aware Routing - Uzaysal ve Trafik Farkındalıklı Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [40]. STAR protokolü diğer pozisyon tabanlı yönlendirme protokollerinden farklıdır. Trafik farkındalığını dikkate alarak hem coğrafi bilgi sisteminden elde ettiği topoloji bilgilerini hem de ortamdaki araçların boyut bilgilerini dikkate alır. SAR

protokolünün eksik yönlerini iyileştirmeyi amaçlamaktadır. Coğrafi bilgi sisteminden aldığı bilgiyi trafik durumunu da dikkate alarak rota tahmini yapmaktadır. Komşu düğümlerin hareketleri gözlemlenir. Beacon mesajları ile komşu düğüm hareketleri de diğer düğümler ile paylaşılır. Bu bilgilere göre trafik yoğunluk tahmini yapmaktadır.

Seet ve diğerleri 2004 yılında yaptıkları çalışmada ASTAR (Anchor based Street and Traffic Aware Routing - Kanca tabanlı Cadde ve Trafik Farkındalıklı Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [41]. ASTAR pozisyon tabanlı bir yönlendirme protokolüdür ve kentsel alanlarda kullanım için tasarlanmıştır. Kaynak ve hedef düğüm arasında bulunan kavşakların sayısını hesaplayabilmek için bölgedeki cadde ve sokak haritalarını kullanır. Paket gönderimi yapılacağı zaman kavşaklarda kesişen yolların trafik durumu göz önünde bulundurularak hesaplama yapmaktadır. Trafik durumu istatistiksel olarak hesaplanır. Belirlenen bir zaman aralığında yollardaki araç sayısı dikkate alınır. ASTAR protokolü kentsel alanlarda kullanıma daha uygun olan, GPRS ve GSR protokollerinden farklı bir yerel arama yöntemi kullanır. Kullandığı yerel arama yönteminde, sokak ve caddeler arasından kanca yol olarak adlandırılan bir yol seçer ve sıçrama yapılacak düğümü bu noktaya göre belirler. Trafik farkındalıklı yapıya sahip olması ve farklı bir lokal arama yaklaşımına sahip olmasından dolayı GPSR ve GSR'ye oranla kentsel alanlarda daha başarılı sonuçlar elde etmektedir. Ancak, yönlendirme işlemini diğer protokollere oranla daha fazla süre kullanarak gerçekleştirebilir.

Kentsel alanlarda çok sayıda yol ve kavşağın olması herhangi bir hedef için bir çok rotanın oluşmasına neden olmaktadır. Mesafe olarak en kısa yol her zaman en uygun yol olmayabilir. Tee ve Lee 2009 yılında yaptıkları çalışmada JARR (Junction based Adaptive Reactive Routing - Kavşak tabanlı Uyarlamalı Tepkisel Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [42]. JARR protokolü kentsel alanlardaki yollarda araç yoğunluğunu dikkate alarak en uygun olan rotaları tespit etmeye çalışan bir yönlendirme protokolüdür. Bir yolun yoğunluk durumu toplanan beacon mesajlarının miktarı ile tespit edilmektedir. Yoğunluk bilgisi araçların hızları ile doğru orantılıdır. Hız bilgisi de bir yolun yoğunluğunu tespit etmek için kullanılmaktadır. JARR protokolü, beacon mesaj miktarı ve araçların hızlarını kullanarak en uygun rotayı tespit etmektedir. Kentsel alanlarda kullanımında başarılı sonuçlar elde etmiştir.

Jerbi ve diğerkleri 2007 yılında yaptıkları çalışmada GyTAR (Greedy Traffic-Aware Routing - Anında Tepkimeli Trafik Farkındalıklı Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [43]. GyTAR kavşak tabanlı bir yönlendirme protokolüdür. Kentsel alanlardaki kavşakları bularak olası rotalar arasından en uygun olanları bulmaya çalışmaktadır. Lokal maksimum probleminin üstesinden gelmek için taşı-gönder yaklaşımını kullanmaktadır. Kavşak noktalarını bulmak için dijital haritalar kullanır. Trafik yoğunluğuna ve hedefe uzaklığa göre kavşaklara puanlar verilmektedir. En yüksek puana sahip olan kavşaktaki araçlar üzerinden paket gönderimi yapılmaktadır.

Kümeleme yaklaşımlarında karşılaşılan en büyük problem kümelerin birbirleri ile iç içe geçmesidir. Yüksek hızdan dolayı sürekli olarak kümeler birbirlerine karışmaktadır. Wang ve Wang 2010 yılında yaptıkları çalışmada TIBCRPH (Traffic Infrastructure Based Cluster Routing Protocol with Handoff – Trafik Altyapı Tabanlı Kümelemeli Yönlendirme Protokolü) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [44]. TIBCRPH protokolü bu problemin üstesinden gelmek için geliştirilmiştir. İki kümenin kesiştiği alanda bulunan araçlar iki kümeye de üyedir. Bu alanda bulunan aracın hız ve yön bilgisi iki küme başı aracın hız ve yön bilgisine bağlı olarak hesaplanır. Bu hesaplama sonucunda daha uygun olan kümeye dahil olunur. Trafik yoğunluğu ve hıza göre karşılaştırıldığında diğerk kümeleme yaklaşımlarına göre daha başarılı sonuçlar alındığı görülmüştür.

Nakom ve Rojviboonchai 2010 yılında yaptıkları çalışmada DECA (Density Aware Reliable Broadcasting - Yoğunluk Farkındalıklı Güvenilir Yayın) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [45]. DECA protokolü rota keşif işlemi yaparken araçların pozisyon bilgisine ihtiyaç duymayan her yöne yayın tabanlı bir yönlendirme protokolüdür. Belirli bir sıçrama sayısı uzaklıktaki alan içerisindeki araç yoğunluğu bilgisini kullanır. Bir düğüm yayın mesajı göndermeden önce, komşuları arasından yoğunluk bilgisi en fazla olan düğümü sıçrama noktası olarak seçer. Diğerk düğümler ise belli bir süre bekleme moduna geçer. Bu süre bitmeden seçilen sıçrama noktası düğümünden herhangi bir yayın mesajı gelmezse, alınmış olan yayın mesajı diğerk düğümlere gönderilir. Alınan her yayın mesajına bir sıra numarası atanır ve beacon mesajları ile düğümler arasında paylaşılır. Böylece, hangi yayın mesajının daha yeni olduğu belirlenir. DACA, esnek bir yönlendirme protokolüdür. Düğümlerin pozisyon bilgisine ihtiyaç duymadığından dolayı, kentsel ya da otoyollarda kuruluna VANET ortamlarında kullanılabilir.

Souza ve diğeri 2013 yılında yaptıkları çalışmada MAV-AODV (Multicast with ant colony optimization for VANET) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [46]. MAV-AODV protokolü karınca koloni algoritmasına dayanan sezgisel bir yönlendirme protokolüdür. Karınca koloni algoritmasında olduğu gibi hedefe yönelik rotalar arasından en uygun olanını bulmak için feromon (karıncaların ürettiği salgı) miktarını kullanır. Keşfedilen rotalar hakkındaki bilgi beacon mesajları kullanılarak diğer araçlara bildirilir. Beacon mesajını araç aldığı zaman, mesajın içerisindeki rota bilgisinin geçerli olacağı süreyi hesaplar. Hedef düğüme Ant-RREQ-J mesajı yayın halinde gönderilir. Bu mesajı alan ara düğümler sıçrama sayısını ve rotanın aktif olduğu zaman bilgisini mesaja ekleyerek hedefe doğru iletir. Hesaplanan bu bilgi kaynak düğüme doğru Ant-RREP mesajı ile cevap olarak gönderilir. Kaynak düğüm, gelen cevaplar arasından en uygun olanı rota olarak seçer ve veri gönderimi için kullanır.

Veri gönderim şekli belirlenmeden ya da başka bir deyişle paket gönderim algoritmasına (topoloji tabanlı, istek tabanlı) karar verilmeden önce araç ile geçiş noktaları arasında uçtan uca rotanın kurulması gerekmektedir. Rotanın tespiti için gerekli olan paket gönderimi, asıl paketten daha fazla daha masraflı olabilir. Buna ek olarak, eğer ağ ayrıkça, uçtan uca rota hiç belirlenemeyebilir. Bundan dolayı, topoloji tabanlı yönlendirme protokolleri coğrafik ve pozisyon tabanlı protokoller kadar etkili değildir. Minimum gecikme ile paketlerin hedefine gönderilmesi için Shafiee ve Leung tarafından CMGR (Connectivity-aware minimum-delay geographic routing - En Az Gecikmeli Coğrafik Yönlendirme) isimli araçsal ağ yönlendirme protokolü geliştirilmiştir [47]. CMGR protokolü coğrafik tabanlı bir algoritmaya dayanmaktadır. Geçiş noktalarına ulaşmak için bir rota oluşturmak isteyen araç, RP isimli bir keşif paketi üretir. Bu paketin içerisinde aracın kendisini tanımlayan kısımlar vardır. Bu paket ağ içerisine yayınlanır. Paketin üretildiği zamandan paketi alan düğüm o an ki zamanı çıkartır. Eğer eşik değerinden fazla ise paket düşürülür. Burada amaç, en az gecikme değerine sahip rotayı kullanarak paketi geçit noktasına ulaştırmaktır. Geçiş noktalarına gelen RD paketleri farklı rotalar üzerinden geldiğinde, geçiş noktasının bunlar arasından en uygununu seçmesi gerekmektedir. Seçilmiş olan rota üzerindeki RD'lere göre tüm kavşaklar belirlenmektedir. RD'ler junctions sequence (JS) olarak isimlendirilmiştir. Geçiş noktası bir RR paketini JS'leri ID'lerini, lokasyonu ve rota isteği yapan aracın hızını kıyaslayarak üretilmiştir. En uygun rotanın seçimi için araç yoğunluğu kullanılmıştır. Yoğunluğu en az olan rota en iyi rota olarak değerlendirilmiştir. Rotalar arasından en az yoğunluğa sahip olan rota seçilmektedir. Beacon mesajları ile yoğunluk tespiti yapılmaktadır. Ancak araç

yoğunluğu oldukça değişkendir. Bu yaklaşımda yoğunluk değişikliği anında tespit edilememektedir [47].

Bitam ve diğerleri 2013 yılında yaptıkları çalışmada HyBR (Hybrid Bio inspired Bee Swarm Routing Protocol - Arı Kolonisinden Esinlenmeli Yönlendirme Protokolü) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir [48]. HyBR protokolü, VANET'lere özgü özellikleri göz önünde bulunduran sürekli öğrenme paradigmasına göre çalışan bir protokoldür. Doğal ortamdaki arıların bir yiyecek buldukları zaman bunu kovandaki diğer arılara nasıl bildirdiği incelenerek modellenmiştir. Arıların birbirleri ile iletişim modeli açıklanmıştır. Arıların yiyecek bulmak için birbirleri arasında kurdukları iletişimden esinlenilerek protokol tasarlanmıştır. VANET ortamının da arıların doğal ortamı gibi görülebileceği ifade edilmiştir. RSU biriminin hedef düğümün ve aradaki düğümlerin bir arı kolonisinde nelere karşılık geldiği belirtilmiştir. Ağ yoğun olduğu zaman topoloji tabanlı yönlendirme, yoğun olmadığı zaman coğrafik tabanlı yönlendirme yapan hibrit bir protokoldür. Temel çalışma mantığı ağın yoğunluğuna göre, yönlendirme algoritmasını değiştirmesidir. Her bir düğümde GPS cihazı olduğu için, ortamdaki tüm araçların pozisyon bilgisi pozisyon tablosu olarak adlandırılan bir tabloda tutulur. Her bir düğümün kendisine ait bir yönlendirme tablosu vardır. Hedeflere farklı rota bilgileri bu tabloda tutulur. Ağın yoğunluğunun nasıl belirleneceği açıklanmıştır. Ağ daha ufak alt ağlara ayrılarak yoğunluk ölçümü kaynak ve hedef arasında belirlenir. Eğer bir alt ağdaki düğüm sayısı önceden belirlenen eşik değerinden daha fazla ise yoğun ağ, değilse ayrık ağ olarak ifade edilmiştir. Yoğun ve ayrık olma durumuna göre de farklı yönlendirme algoritması çalıştırılmaktadır [48].

Kümeleme yaklaşımları kullanılarak farklı birçok ağ türü için protokoller geliştirilmiştir. Standart kümeleme algoritmalarının dezavantajlarını ortadan kaldırabilmek için pasif kümeleme olarak adlandırılan yeni kümeleme yaklaşımı geliştirilmiştir. MANET'ler için geliştirilen ve pasif kümeleme mantığına dayalı bir çok protokol araştırmacılar tarafından geliştirilmiştir. Ancak, halihazırda MANET'ler için geliştirilen protokoller VANET ortamında istenildiği verimlilikte çalışmamaktadır. Wang ve Lin 2013 yılında yaptıkları çalışmada VANET'ler için yeni bir pasif kümeleme tabanlı yönlendirme protokolü geliştirmişlerdir ve PassCAR (Passive Clustering Aided Routing Protocol - Pasif Kümeleme Yardımlı Yönlendirme Protokolü) olarak adlandırmışlardır. Pasif kümeleme yaklaşımı geleneksel kümeleme yaklaşımına göre paket yönlendirme yükünü azaltmıştır. Araçların

davranışlarını ve bağlantı kalitesini dikkate alan metrikler geliştirilerek protokolde kullanılmıştır [49].

Gerçek ortamlarda trafik şekilleri dikkate alınması gereken bir durumdur. Trafik ışıkları ve yoğunluk şekillerinin modellenmesi gerçekçi bir yaklaşım ortaya koyar. Bu durumları dikkate alan bazı protokoller literatürde bulunmaktadır. Sürekli olarak akıp giden trafikte kümelenmelerinin olması çok olasıdır. Bu boşlukları dikkate almaya çalışan bazı yaklaşımlar literatürde bulunmaktadır. Caddelerin ve sokakların varlığından haritaların önceden yüklenmesi ile haberdar hale getirilebilir. Bernsen ve Manivannan 2012 yılında yaptıkları çalışmada RIVER (Reliable Inter Vehicular Routing Protocol - Araçlar Arası Güvenilir Yönlendirme Protokolü) isimli araçsal ağ yönlendirme protokolü geliştirmişlerdir. RIVER protokolü, ağda dolaşan paketlerin içeriklerini inceleyerek pasif olarak caddelerdeki yoğunluk hakkında bilgi sahibi olur. Ağdaki düğümler paketleri toplayarak kendi çevresindeki caddeler hakkında güvenilirlik durum bilgisi oluşturur. Bu caddeler bir liste şeklinde tutulur. Bu listeye göre cadde seçimi yapılır. En az değere sahip cadde en güvenilirdir. Eğer aynı güvenilirlik değerine sahip iki cadde varsa, daha kısa olan cadde seçilir [50]. Araç yoğunluğundan dolayı ortaya çıkabilecek sorunları çözmeye yönelik geliştirilen yönlendirme protokolleri Çizelge 2.3’de farklı özelliklere göre karşılaştırmalı olarak sunulmuştur.

Çizelge 2.3. Araç Yoğunluğu Odaklı Araçtan Araca Yönlendirme Protokolleri

Protokoller	Yönlendirme Algoritması	Harita Kullanımı	GPS Kullanımı	Araç Yoğunluğu	Taşı-Gönder Desteği	İletişim Ortamı
ASTAR	Greedy	Var	Var	Orta	Yok	Kentsel
STAR	Greedy	Var	Var	Orta	Yok	Kentsel
JARR	Greedy	Var	Var	Çok	Yok	Kentsel
GyTAR	Greedy	Var	Var	Orta	Yok	Kentsel
CMGR	Greedy	Yok	Var	Çok	Yok	Kentsel
TIBCRPH	Çoklu Sıçrama	Var	Yok	Çok	Var	Kentsel
DECA	Çoklu Sıçrama	Yok	Yok	Çok	Var	Otoyol
MAV-AODV	Çoklu Sıçrama	Yok	Var	Orta	Yok	Kentsel
HyBR	Çoklu Sıçrama	Yok	Var	Çok	Var	Kentsel
PassCAR	Çoklu Sıçrama	Yok	Yok	Orta	Var	Kentsel
RIVER	Çoklu Sıçrama	Var	Yok	Çok	Var	Kentsel

Literatürde araçsal ağlara yönelik yapılan çalışmaları farklı özelliklere göre sınıflandıran birçok çalışma bulunmaktadır [51,52]. Veri iletişim tabanlı çalışan ve yüksek hızlara ihtiyaç duyan birçok uygulama ve teknoloji ortaya çıkmaktadır. Bu uygulama ve teknolojilerin verimli şekilde kullanılabilmesi, veri transfer hızı ile doğrudan bağlantılıdır. Özellikle 5G teknolojisinin günlük hayata girmesiyle birçok yeni uygulama ve teknolojinin ortaya çıkması muhtemeldir. Araçsal ağların ise, ortaya çıkacak bu yeni teknolojilerin başını çekmesi beklenmektedir. Bunun sonucunda, araçsal ağların bir kısmını oluşturan V2I iletişiminin öneminin artmasına neden olacaktır. Literatürde V2I iletişimine yönelik araştırmalar ve çalışmalar yapılmaya başlanmıştır [53]. Bununla beraber V2V iletişimi, cihazdan cihaza iletişim olarak daha detaylı şekilde literatürde sınıflandırılmaya başlanmıştır [54,55]. Gerçek zamanlı uygulamaların VANET ortamında kullanılabilmesi birçok yeniliğin gündelik hayatta kullanılabilmesine imkan tanıyacaktır. Gerçek zamanlı uygulamaların VANET ortamındaki verimliliğini ortaya çıkartmaya yönelik çalışmalar bulunmaktadır [56]. Araçsal ağlarda verinin yayılımından kaynaklanan birçok problem ortaya çıkmaktadır. Bunların başında, aynı verilerin tekrarlı olarak düğümler arasında dolaştırılmasıdır. Verilerin tekrarlı şekilde araçsal ağ ortamında dolaştırılması daha fazla kaynak tüketilmesi ve iletişim kanalının gereksiz yere meşgul edilmesi gibi performans sorunlarının ortaya çıkmasına neden olmaktadır. Literatürde verilerin araçsal ağ ortamında yayılmasının kontrollü şekilde yapılması ve tekilleştirilmesine yönelik birçok çalışma bulunmaktadır [57-59]. Araçsal ağların bulut teknolojiler ile tümleşik hale getirilerek daha etkin şekilde kullanılmasını öngören çalışmalar bulunmaktadır [60].

3. VNPER PROTOKOL TASARIMI

Ağdaki düğümler arasında verilerin bir noktadan başka bir noktaya gönderilmesi sırasında hangi düğümlerin kullanılacağı, hangi yolların veri iletimi için seçileceği yönlendirme protokolü tarafından belirlenir. Paketin hedefine kabul edilebilir bir gecikme ile gönderilmesi, paket kaybının en az olacak şekilde iletimin gerçekleştirilmesi, yönlendirme işleminin düğümlere en az yük getirecek şekilde yapılması bir yönlendirme protokolünden genel olarak beklenen yeterliliklerdir. Araçtan araca VANET iletişim modeli için bir yönlendirme protokolü tasarlamak oldukça zordur. Çünkü V2V ortamında topoloji değişimi çok hızlı olmaktadır. Araç yoğunluğunun fazla ya da az oluşu verilerin hedefine kabul edilebilir seviyelerde ulaştırılmasını zorlaştırmaktadır. Bu çalışmada, V2V ağlar için bir yönlendirme protokolü geliştirilmiştir. Geliştirilen protokol VNPER (Vehicular Next Position Estimation Routing-Araçlar için Gelecek Rota Tahmin tabanlı Yönlendirme) olarak isimlendirilmiştir. Araçların hızları ve hareket tahminine dayalı olarak geliştirilmiş olan sezgisel ve ölçeklenebilir bir yönlendirme protokolüdür.

3.1. VNPER Protokolünün Paket ve Zamanlayıcı Tipleri

Herhangi bir ağ ortamında verilerin kaynaktan hedefine ulaştırılabilmesi yönlendirme protokolleri sayesinde gerçekleştirilir. Yönlendirme protokolleri, hedef düğümü tespit etmek ve hedef düğüme ulaşırken hangi yolların tercih edileceğini belirlemek için ağ ortamında birçok farklı paket tipini düğümler arasında dolaştırır. Bu paketler yönlendirme protokolü tarafından üretilir ve hem hedef düğümün hem de rotaların tespit edilmesi için kullanılır. Yönlendirme protokolünün kalitesi ise üretmiş olduğu paket tipi ve paket miktarı ile doğru orantılıdır. VNPER protokolünde kullanılmak üzere tasarlanan paket ve zamanlayıcı tipleri aşağıda belirtilmiştir.

3.1.1. Rota keşif paketi

Araçsal ağ ortamında veri gönderiminin gerçekleştirilebilmesi için, kaynak düğümün hedef düğüme nasıl ulaşacağını bilmesi gerekmektedir. Hedefin bulunabilmesi için bir keşif yapılması gerekmektedir. VNPER protokolünde bu işlemin yapılabilmesi için RDP (Route Discovery Packet-Rota Keşif Paketi) olarak isimlendirilen bir paket tipi oluşturulmuştur. Bu

paket tipi, hedef düğümün keşfi sırasında kullanılmaktadır. Düğümler arasında RDP paketleri dolaştırılarak hedefin keşfedilmesi sağlanır. RDP paketinin yapısı Şekil 3.1’de gösterilmiştir.

RDP paketinin içerisinde 8 farklı alan bulunmaktadır. Benzetim ortamında kullanılmak üzere tasarlanan nsaddr_t tipindeki değişkenlerde düğümlerin numaraları tutulmaktadır ve bellekte 1 bayt yer kaplamaktadır. Rdp_src değişkeninde RDP paketini üreten düğümün adresi, rdp_dst değişkeninde ise RDP paketinin ulaştırılmak istendiği düğümün adresi tutulmaktadır. RDP paketinde 3 adet u_int32_t tipinde değişkenin tutulduğu alan bulunmaktadır. Kaynak düğüm tarafından üretilen her bir RDP paketine bir sıra numarası verilir ve rdp_src_seqno değişkeninde saklanır. Bunun dışında her bir hedef düğüm için üretilen RDP paketine de bir sıra numarası verilir ve rdp_dst_seqno değişkeninde saklanır. RDP paketleri ağ ortamına broadcast olarak gönderilir ve her broadcast yayını için pakete bir id numarası verilir. Rdp_bcast_id kısmında paketlere verilen id numarası saklanır. U_int_32_t tipindeki değişkenler bellekte 4 bayt yer kaplamaktadır. RDP paketini alan her düğüm paketin rdp_hop_count alanındaki değeri bir artırarak komşularına gönderir. Bu sayede, her bir RDP paketinin kaynaktan hedefe ulaşırken kaç adet düğümden sıçrayarak hedefine ulaştığı tespit edilebilir. Rdp_hop_count değişkeninde kaynak ile hedef arasında rota üzerindeki düğüm sayısı tutulmaktadır. Her bir RDP paketine üretildiği andaki zaman bilgisi eklenir. Bu bilgi rdp_timestamp değişkeninde tutulur ve double tipinde bir değişkendir. Paketin üretim tarihini ifade eder. Paketin tipi ise pkt_type değişkeninde tutulmaktadır. RDP paketinin içerisindeki bu alanlar hedef düğüm tespiti sırasında çeşitli amaçlar için kullanılmak üzere tasarlanmıştır. RDP paketinin toplam boyutu 30 bayttır.

nsaddr_t rdp_src	u_int32_t rdp_src_seqno	u_int8_t rdp_hop_count	u_int32_t rdp_bcast_id
nsaddr_t rdp_dst	u_int32_t rdp_dst_seqno	u_int8_t pkt_type	double rdp_timestamp

Şekil 3.1. RDP paketinin başlık yapısı

3.1.2. Rota teklif paketi

Kaynak düğümden hedefe doğru gönderilen keşif paketlerini alan düğümlerin kaynak düğüme cevap olarak gönderdiği paket tipidir. Bu paketler sayesinde RDP paketlerine yanıt

verilir ve hedefin kaynak düğüme bilgi vermesi için kullanılır. Tek yönlü bir iletişim söz konusudur. ROP (Route Offer Packet-Rota Teklif Paketi) paketini alan her düğüm, ROP paketini gönderen düğüme ulaşabilecek rotaları öğrenmiş olur. Aynı düğüm farklı ROP paketleri alırsa, bunlardan en iyi rota bilgisine sahip olanı yönlendirme tablosuna aktarırken, diğer rotaları topoloji tablosuna aktarır. ROP paketinin yapısı Şekil 3.2’de gösterilmiştir.

ROP paketinin içerisinde 7 farklı alan bulunmaktadır. Nsaddr_t tipindeki değişkenlerde düğümlerin numaraları tutulmaktadır ve bellekte 1 bayt yer kaplamaktadır. Rop_src değişkeninde ROP paketini üreten düğümün adresi, rop_dst değişkeninde ise ROP paketinin ulaştırılmak istendiği düğümün adresi tutulmaktadır. Her bir hedef düğüm için üretilen ROP paketine bir sıra numarası verilir ve rop_dst_seqno değişkeninde saklanır. Her bir ROP paketinin hedefe ulaşırken kaç adet düğüm üzerinden geçtiği bilgisi rop_hop_count değişkeninde saklanır. Her bir ROP paketine üretildiği andaki zaman bilgisi eklenir. Bu bilgi rop_timestamp değişkeninde tutulur. ROP paketinin benzetim ortamı içerisinde ne zaman yok edileceği bilgisi ise rop_lifetime değişkeni içerisinde tutulur. Paketin tipi ise pkt_type değişkeninde tutulmaktadır. ROP paketinin içerisindeki bu alanlar RDP paketine cevap verilmesi sırasında kullanılmak üzere tasarlanmıştır. ROP paketinin toplam boyutu 30 bayttır.

nsaddr_t rop_src	u_int8_t pkt_type	u_int32_t rop_dst_seqno	double rop_timestamp
nsaddr_t rop_dst	u_int8_t rop_hop_count	double rop_lifetime	

Şekil 3.2. ROP paketinin başlık yapısı

3.1.3. Hello paketi yapısı

VANET ortamında düğümlerin kendi komşularını bilmesi gerekmektedir. Düğümler sabit olmadıkları için komşuluklar sürekli olarak değişmektedir. Bu hareketlilik sürekli olarak hello paketlerinin değişimi ile takip edilebilir. İki düğüm arasında komşuluk kurulması için kullanılan paketlerdir. Bu paketlerin komşu düğümler arasında karşılıklı değiştirilmesi sonucu, birbirlerinin hızlarını, yönünü ve pozisyonlarını öğrenirler. Bu bilgiler daha sonra kullanılmak amacıyla komşuluk tablosunda tutulmaktadır. Hello paketinin yapısı Şekil 3.3’de gösterilmiştir.

Hello paketinin içerisinde 10 farklı alan bulunmaktadır. Nsaddr_t tipindeki değişkenlerde düğümlerin numaraları tutulmaktadır ve bellekte 1 bayt yer kaplamaktadır. hi_src değişkeninde Hello paketini üreten düğümün adresi, hi_dst değişkeninde ise Hello paketinin ulaştırılmak istendiği düğümün adresi tutulmaktadır. Paket tipi, pkt_type değişkeni içerisinde saklanmaktadır. Hello paketi sadece komşu düğümlere gönderilmektedir. Hello paketini alan düğüm, paketin hi_hop_count alanına bakarak diğer düğümlere yanlılıkla paketin iletilmesini engeller. Bu değişken içerisindeki değer 0'dan büyükse paket yok edilir. Paketin üretildiği sıradaki düğümün hızı hi_spd değişkeninde, X koordinat pozisyonu hi_posX değişkeninde, Y koordinat pozisyonu hi_posY değişkeninde, düğümün X koordinatına göre ivmesi hi_veloX değişkeninde, düğümün Y koordinatına göre ivmesi hi_veloY değişkeninde saklanmaktadır. Her bir Hello paketinin üretildiği andaki zaman bilgisi hi_timestamp değişkeninde tutulur ve hello paketin üretim tarihini ifade eder. Hello paketinin içerisindeki bu alanlar, düğümler arası komşuluk kurulması ve komşu düğümler hakkında detay bilgilerin elde edilmesi sırasında kullanılmak üzere tasarlanmıştır. Hello paketinin toplam boyutu 54 bayttır.

nsaddr_t hi_src	u_int8_t pkt_type	u_int8_t hi_spd	double hi_posY	double hi_veloY
nsaddr_t hi_dst	u_int8_t hi_hop_count	double hi_posX	double hi_veloX	double hi_timestamp

Şekil 3.3. Hello paketinin başlık yapısı

3.1.4. Protokol işleyiş zamanlayıcıları

VNPER protokolünün çalışması sırasında bazı paket tiplerinin belirli zaman aralıklarında tekrarlı bir şekilde gönderilmesi gerekmektedir. Bazı paketler ise, belirli bir süre geçtikten sonra ağda dolaşıyor ise yok edilmesi gerekmektedir. Bazı durumlarda da komşuluk, ileri komşuluk, yönlendirme ve topoloji tablosunda bulunan kayıtlardan zaman aşımına uğramış olanların silinmesi gerekmektedir. Zaman aşımının olup olmadığının belirlenmesi için, zamanlama işlemini gerçekleştiren yapılara ihtiyaç duyulmaktadır. VNPER protokolünde 6 tane zamanlayıcı yapısı bulunmaktadır. Bu zamanlayıcılar; anycast, hello, neighborhood, nextneighborhood, rtcache ve tpcache'dir. VNPER protokolünde kullanılan zamanlayıcılar Şekil 3.4'de gösterilmiştir.

AnycastTimer

PROTO	Event
*agent	intr
AnycastTimer(Proto* a : agent (a))	
void handle(Event*)	

HelloTimer

PROTO	Event
*agent	intr
HelloTimer(Proto* a : agent (a) {})	
void handle(Event*)	

NeighborhoodTimer

PROTO	Event
*agent	intr
NeighborhoodTimer(Proto* a : agent (a))	
void handle(Event*)	

RTCachetimer

PROTO	Event
*agent	intr
RTCachetimer (Proto* a : agent (a) {})	
void handle(Event*)	

NextNeighborhoodTimer

PROTO	Event
*agent	intr
NextNeighborhoodTimer(Proto* a : agent (a))	
void handle(Event*)	

TPCachetimer

PROTO	Event
*agent	intr
TPCachetimer (Proto* a : agent (a) {})	
void handle(Event*)	

Şekil 3.4. Protokülün işleyişi sırasında kullanılan zamanlayıcılar

VANET ortamındaki komşu düğümlerin tespit edilmesi veya hedef düğümün bulunması gibi ihtiyaçlardan dolayı ağ ortamına broadcast olarak rdp veya hello paketlerini gönderir. Her bir broadcast gönderimi için düğümün kendisi bir tablo tutar. Bu tablo içerisinde zaman aşımına uğramış kayıtların bulunup silinmesi gerekmektedir. Bu işlemin yerine getirilmesi için belli periyotlarda tablonun içerisindeki gereksiz kayıtların silinmesi gereklidir. Anycast Timer zamanlayıcısı sayesinde bu işlem yerine getirilir. VANET ortamında iki düğümün komşu olması için birbirlerine doğrudan erişebilir halde olması gereklidir. Komşuluk ilişkisi düğümlerden birinin hello paketini göndermesi ile başlar. Her düğüm yeni bir komşusunun olup olmadığını tespit edilmesi ya da mevcut komşusunun durumunda bir değişiklik olup olmadığını belirlenmesi için belirli periyotlarda kapsama alanı içerisindeki düğümlere hello paketi gönderir. Hello timer zamanlayıcısı sayesinde belirli periyotlarda komşulara hello paketinin gönderilmesi sağlanır. VANET ortamındaki düğüm hareketliliği çok hızlı olduğundan dolayı, düğümlerin komşuluk durumları sürekli olarak değişmektedir. Düğümlerin komşu düğümlerin durumunu bilmesi gerekmektedir. Her düğüm komşuluk tablosu olarak isimlendirilen bir tablo tutar ve bu tabloda komşu düğümler ile ilgili kayıtlar

bulunur. Neighborhood timer zamanlayıcısı sayesinde komşuluk tablosunda zaman aşımına uğramış gereksiz kayıtların silinmesi sağlanır. Belirli periyotlarda komşuluk tablosu içerisindeki kayıtların temizlenmesi neighborhood timer zamanlayıcısı sayesinde gerçekleştirilir. VNPER protokolünde ileride komşu olması tahmin edilen düğümler ileri komşuluk tablosu olarak isimlendirilen bir tabloda tutulmaktadır. Tıpkı komşuluk tablosunda olduğu gibi, bu tablonun içerisinde bulunan kayıtlardan zaman aşımına uğramış olanların temizlenmesi gerekmektedir. İleri komşuluk tablosundaki bu işlem, nextneighborhood zamanlayıcısı ile gerçekleştirilmektedir. VANET ortamındaki her düğümde bir yönlendirme tablosu bulunmaktadır. Yönlendirme tablosunda herhangi bir hedef düğüme ait rota kayıtları tutulmaktadır. Bu tabloya bakarak düğümler hedefine nasıl ulaşacağını bilmektedir. Bu tabloda zaman aşımına uğramış gereksiz bir kaydın bulunması paketlerin yanlış yere gönderilmesine ya da düşürülmesine neden olmaktadır. Bu yüzden, yönlendirme tablosundaki bu kayıtların bulunup temizlenmesi gerekmektedir. RTCache zamanlayıcısı sayesinde, belirli periyotlarda yönlendirme tablosundaki zaman aşımına uğramış kayıtların temizlenmesi sağlanır. Yönlendirme tablosuna ek olarak, herhangi bir hedef için yedek rotalar topoloji tablosu olarak isimlendirilen bir tabloda tutulmaktadır. Tıpkı yönlendirme tablosunda olduğu gibi, topoloji tablosunda da zaman aşımına uğramış ya da gereksiz olan kayıtların bulunup temizlenmesi gerekmektedir. TPCache zamanlayıcısı sayesinde, belirli periyotlarda topoloji tablosunda bulunan zaman aşımına uğramış kayıtların temizlenmesi sağlanır. Özet olarak, bu zamanlayıcılar sayesinde protokolün devamlılığını sürdürebilmesi amacıyla belirli periyotlarda benzer işlemlerin yapılması sağlanmaktadır.

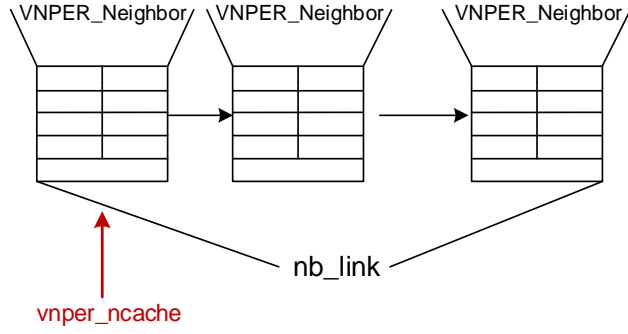
3.2. VNPER Protokolünde Kullanılan Tablolar

VNPER protokolünde kullanılmak için geliştirilen 4 farklı tablo bulunmaktadır. Bu tablolar komşuluk tablosu, ileri görüşlü komşuluk tablosu, yönlendirme tablosu ve topoloji tablosudur. Bu tablolar farklı amaçlar için kullanılmaktadır. C++'da sınıf olarak türetilmiştir. Bağlı listeler kullanılarak oluşturulan nesnelere birleştirilmiştir. Komşuluk tablosundaki nesnelere, kaynak düğüm ve komşu düğümün adresi, paket tipi, sıçrama sayısı, komşu düğümün hızı, komşu düğümün X ve Y koordinatları, komşu düğümün X ve Y yönündeki ivmeleri ve zaman aşımına uğrayacağı zaman bilgisinin tutulduğu 10 alan bulunmaktadır. Şekil 3.5'de komşuluk tablosunu oluşturulan nesnenin yapısı ve bağlı liste hali gösterilmiştir.

Class VNPER_Neighbor

ns_addr_t hi_src	ns_addr_t hi_dst
u_int8_t pkt_type	u_int8_t hi_hop_count
u_int8_t hi_spd	double hi_posX
double hi_posY	double hi_veloX
double hi_veloY	double nb_expire

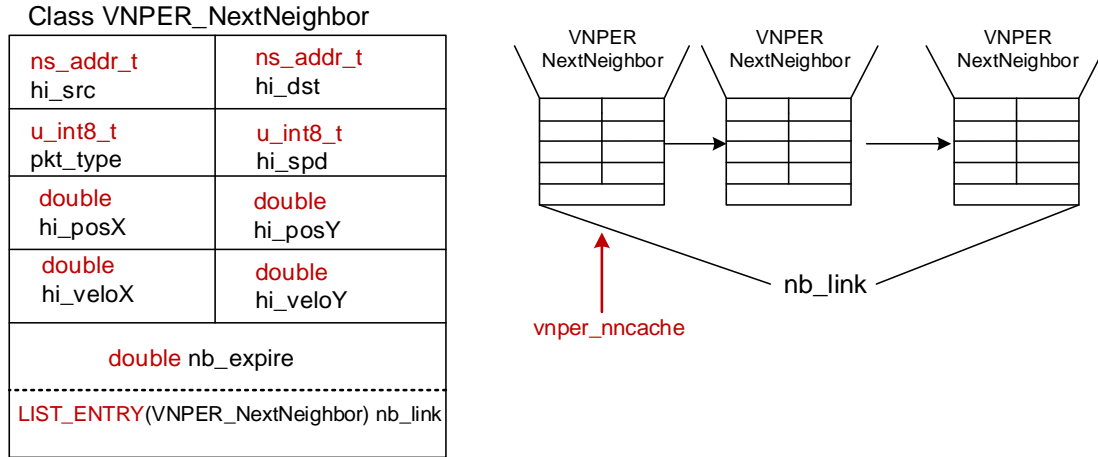
LIST_ENTRY(VNPER_Neighbor) nb_link	



Şekil 3.5. Komşuluk tablosu ve tabloyu oluşturan nesnelere

Bağlı liste şeklinde bellekte saklanan bu tablo sayesinde komşu düğümler hakkında detaylı bilgiye sahip olunur. Bu bilgiler başta bir VANET otamındaki herhangi bir hedef düğüme ait rota kurulması gibi birçok farklı amaç için kullanılmaktadır. Bağlı liste kullanılmasının sebebi dinamik bir yapıya sahip olması ve üzerinde ekleme ve çıkarma işleminin hızlıca yapılabilmesidir. VNPER_neighbor tipinde vnper_ncache olarak adlandırılan işaretçi sayesinde bağlı liste içerisinde dolaşarak zaman aşımına uğramış gereksiz bir kayıt kolaylıkla bulunabilmektedir.

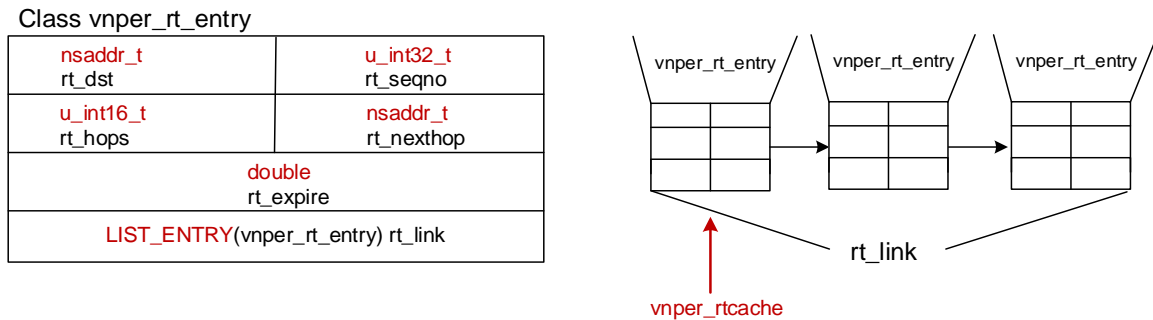
İleri görüşlü komşuluk tablosundaki nesnelere, gelecekte komşuluk kurulması muhtemel düğümler hakkında bilgiler tutulmaktadır. Bu tabloda, kaynak düğüm ve bir sonraki zaman diliminde komşuluk kurulması muhtemel olan düğümün adresi, paket tipi, komşuluk kurulması muhtemel olası komşu düğümün hızı, X ve Y koordinatları, X ve Y yönündeki ivmeleri ve zaman aşımına uğrayacağı zaman bilgisinin tutulduğu 9 alan bulunmaktadır. Şekil 3.6'da ileri görüşlü komşuluk tablosunu oluşturulan nesnenin yapısı ve bağlı liste hali gösterilmiştir.



Şekil 3.6. İleri görüşlü komşuluk tablosu ve tabloyu oluşturan nesnelere

İleri görüşlü komşuluk tablosu da bağlı liste şeklinde bellekte saklanmaktadır. Bu tablo sayesinde bir sonraki zaman diliminde komşu olması muhtemel düğümler hakkında detaylı bilgiye sahip olunur. Bu tablodaki bilgiler sayesinde VANET ortamındaki gelecekte olası komşuluklar tahmin edilebilir ve bunun sonucunda VANET ortamının gelecekteki topolojisi çıkartılabilir. Bu topolojiye göre de hedef düğümler için rotalar belirlenebilir. Her seferinde yeniden rota keşif işlemlerinin yapılmasının önüne geçilmiş olur. Böylece rota keşif işleminin getirmiş olduğu maliyet en aza indirilir. VNPER_Nextneighbor tipinde vnper_nncache olarak adlandırılan işaretçi sayesinde bağlı liste içerisinde dolaşarak zaman aşımına uğramış gereksiz bir kayıt kolaylıkla bulunabilmektedir.

Yönlendirme tablosundaki nesnelere, hedef düğümün adresi, kayıt numarası, sıçrama sayısı, bir sonraki sıçrama noktası, rotanın zaman aşımı olmak üzere 5 tane alan bulunmaktadır. Şekil 3.7’de yönlendirme tablosunu oluşturan nesnenin yapısı ve bağlı liste hali gösterilmiştir.



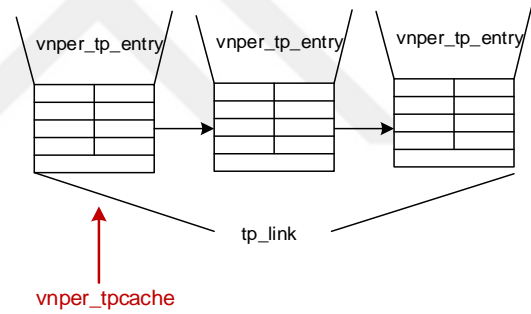
Şekil 3.7. Yönlendirme tablosu ve tabloyu oluşturan nesnelere

Bu tabloda VANET ortamındaki hedef düğümler için kayıtlar tutulmaktadır. Bir hedefe ulaşmak için birden fazla rota olması olası bir durumdur. Bu rotalar arasından en az maliyete sahip olan rotalar yönlendirme tablosunda saklanmaktadır. Bu tabloya bakarak düğümler hedef düğüme nasıl ulaşacağını bilmektedir. Bu tablo bağlı liste şeklinde bellekte tutulmaktadır. Sürekli olarak bu tablodan kayıtlar silinmekte, eklenmekte ya da güncellenmektedir. Dinamik bir yapıya sahiptir. Vnper_rt_entry tipinde vnper_rtcache olarak adlandırılan işaretçi sayesinde yönlendirme tablosunun içerisinde ilgili rotanın bulunarak gerekli olan işlemlerin yapılması sağlanmaktadır.

Topoloji tablosundaki nesnelere, hedef düğümün adresi, kayıt numarası, sıçrama sayısı, bir sonraki sıçrama noktası, rotanın zaman aşımı, flags, priority, tablo içi sıra numarası ve reserve olmak üzere 9 tane alan bulunmaktadır. Şekil 3.8’de topoloji tablosunu oluşturan nesnenin yapısı ve bağlı liste hali gösterilmiştir.

Class vnper_tp_entry

nsaddr_t tp_dst	u_int32_t tp_seqno
u_int16_t tp_hops	nsaddr_t tp_nexthop
double tp_expire	u_int8_t tp_flags
int tp_priority	double tp_seq_queue
int reserved	
LIST_ENTRY(vnper_tp_entry)tp_link	



Şekil 3.8. Topoloji tablosu ve tabloyu oluşturan nesnelere

Topoloji tablosunda VANET ortamındaki hedef düğümler için yedek rota kayıtları tutulmaktadır. Araçsal ağlarda belirli bir hedefe farklı birçok rota üzerinden ulaşılabilir. Yönlendirme tablosunda en az maliyete sahip rotanın kaydı tutulur. Bunun dışında en az maliyete sahip olan 5 rota ise topoloji tablosunda tutulmaktadır. Bu yedek rotalar sayesinde, yönlendirme tablosunda tutulan asıl rota ile ilgili herhangi bir sıkıntı olması durumunda yeniden rota keşfi yapmak yerine topoloji tablosundaki rotalardan en az maliyete sahip olan rota, yönlendirme tablosuna asıl rota olarak gönderilir ve kullanılmaya başlanır. Topoloji tablosunda tutulacak yedek rota sayısı, yapılan testler sonucunda alınan sonuçlara göre belirlenmiştir. Bundan dolayı, 5 adet rotanın yedek olarak tutulmasına karar verilmiştir. Daha fazla yedek rota tutulması, topoloji tablosunda gereksiz yere büyüme meydana getirerek en az maliyetli rotanın yönlendirme tablosuna aktarılma süresini artırmaktadır. Bu

durum VNPER protokolünün performansını olumsuz yönde etkilemektedir. Daha az yedek rotanın tutulması ise, bazı durumlarda uygun rotanın topoloji tablosunda bulunamamasına neden olmaktadır. Bunun sonucunda ise yeniden rota keşfi yapılmasına gerek duyulmaktadır. Topoloji tablosunda bulunan tp_priority alanı sayesinde, her bir rotaya yönlendirme tablosuna geçme önceliği verilir. Tp_priority değeri en yüksek olan rota, yönlendirme tablosuna geçme önceliğine sahip olan rotadır. Herhangi bir hedef düğüm için topoloji tablosunda 5 adet yedek rota tutulmaktadır. Bu rotalar öncelik değerlerine göre sırası ile ardışık olarak sıralanmaktadır. Sıralama işleminin yapılabilmesi için, tp_seq_queue alanı kullanılmaktadır. Bu tablo bağlı liste şeklinde bellekte tutulmaktadır. Topoloji tablosu, yönlendirme tablosuna göre çok daha dinamik bir yapıya sahiptir. Bundan dolayı, tablodan kayıt silme, ekleme ya da güncelleme işlemleri yönlendirme tablosuna göre daha fazla yapılmaktadır. Tablo içerisinde kayıtların hangilerinin geçersiz olduğunun daha hızlı tespit edilebilmesi için, tp_flag alanı kullanılmaktadır. Tp_flag alanın 1 olması ilgili kaydın geçersiz olduğu anlamına gelmektedir. Sadece bu alandaki değere bakarak daha hızlı ve az maliyet ile geçerliliğini yitirmiş kayıtların bulunması sağlanır. Vnper_tp_entry tipinde vnper_tpcache olarak adlandırılan işaretçi sayesinde topoloji tablosunun içerisinde ilgili rotanın bulunması ve gerekli olan işlemlerin yapılması sağlanmaktadır. Topoloji tablosundaki kayıt, yönlendirme tablosuna en iyi rota olarak gönderilirken topoloji tablosuna özgü olan tp_flag, tp_priority, tp_seq_queue ve reserve alanları çıkartılarak yönlendirme tablosunun formatına uygun hale getirilir. Bu sayede, en iyi yedek rotanın topoloji tablosundan yönlendirme tablosuna aktarılması sağlanmaktadır.

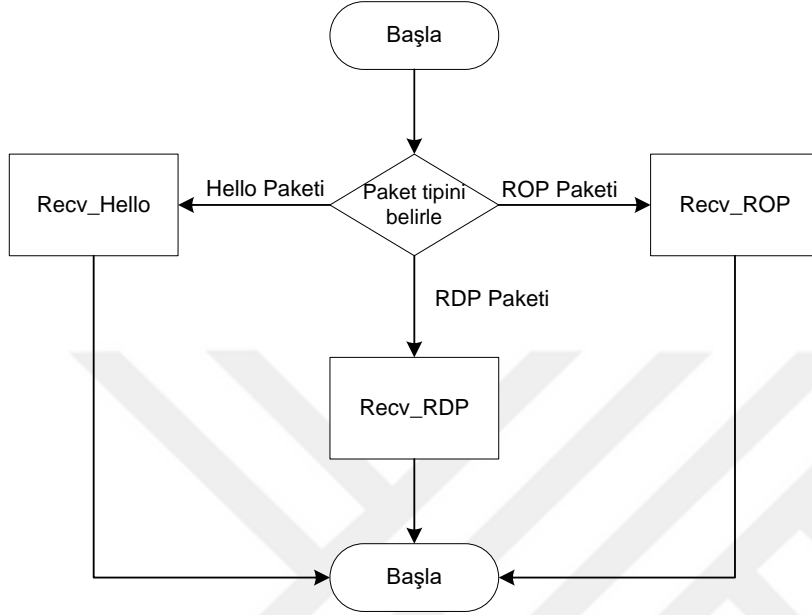
3.3. VNPER Protokolünün Algoritması

V2V iletişimi için geliştirilmiş olan VNPER protokolünün her bir aşamasındaki akış şemaları ve algoritması gösterilmiştir. Kentsel alanlarda kullanım için tasarlanmış bir araçsal ağ yönlendirme protokolüdür. Araçların hızları ve hareket tahminine dayalı olarak geliştirilmiş olan sezgisel ve ölçeklenebilir bir yönlendirme protokolüdür.

3.3.1. VNPER protokolünün paket alım akış şeması

VANET ortamındaki bir düğüme herhangi bir paket geldiği zaman, paketin tipinin bilinmesi gerekmektedir. Paket tipine göre, protokollerin farklı tepkiler vermesi gerekir. Bu yüzden, ilk aşama paket tipinin belirlenip ona göre işleme devam edilmesidir. Şekil 3.9'da bu adımın

akış şeması gösterilmiştir. Geliştirilen yönlendirme protokolünde bir düğüm 3 farklı paket alabilir. Bu paketler hello, RDP ve ROP paketleri olabilir. Alınan paket tipine göre, akış şeması dallanmaktadır. Bu kısım algoritmanın başlangıcı olarak kabul edilmektedir.

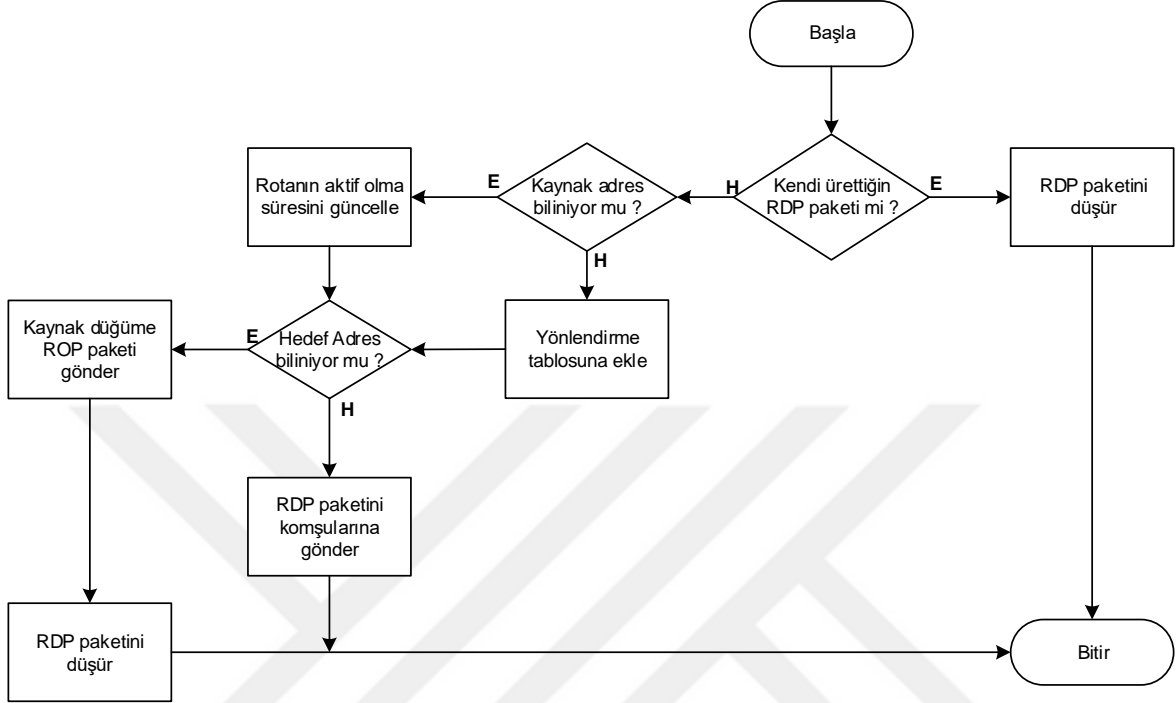


Şekil 3.9. VnPER protokolünün paket alım akış şeması

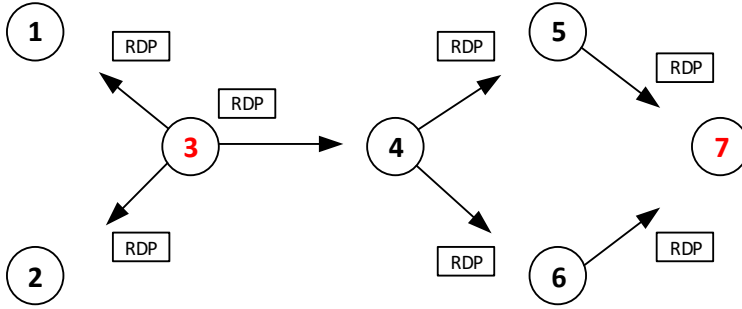
3.3.2. VnPER protokolün rota keşfi akış şeması

Bu kısımdaki algoritma, herhangi bir düğümün rota keşfi yapması gerekli olduğu zaman çalışmaktadır. Algoritmanın başlangıcında düğüme gelen paket RDP ise, Recv_RDP akış şemasına dallanmaktadır. Şekil 3.10'da bu akış şeması gösterilmiştir. Düğüm RDP paketi aldığı anda ilk yapılacak kontrol, bu paketin kendisi tarafından üretilip üretilmediğinin tespit edilmesidir. Kendisinin ürettiği paket, ağdaki bir döngüden dolayı tekrar kendisine gelmiş olabilir. Bu tarz paketlerin tespit edilmesi ağın kullanılabilir olması açısından oldukça önemlidir. Düğümün kendisinin ürettiği bir paket ise, bu paket düşürülür. Değilse, bir sonraki aşamaya geçilir. RDP paketinin kaynak adresinin, paketi alan düğüm tarafından bilinmesi gerekmektedir. Çünkü, bu pakete bir cevap gelmesi durumunda kaynak düğüme nasıl gidileceğini bilmesi gerekmektedir. Eğer bilinmiyorsa, kendisine RDP paketini gönderen komşusunu hedefe ulaşmak için gerekli olan rota olarak kabul eder ve yönlendirme tablosuna ekler. Eğer biliyorsa, öncelikle rotanın aktif olma süresini tazeler. Daha sonra hedef adresin bilinip bilinmediği sorgulanır. Hedef kendisi ise ya da biliyorsa, kaynak düğüme ROP paketi gönderir ve almış olduğu RDP paketini düşürür. Eğer hedef adres bilinmiyorsa, almış olduğu RDP paketini komşularının sayısı kadar kopyalayarak her bir

komşusuna iletir. Alan komşu düğümlerde aynı algoritma ile hedef düğüm aranır. RDP paketlerinin düğümler arasında yayılması Şekil 3.11’de gösterilmiştir.



Şekil 3.10. Rota keşif algoritmasının akış şeması



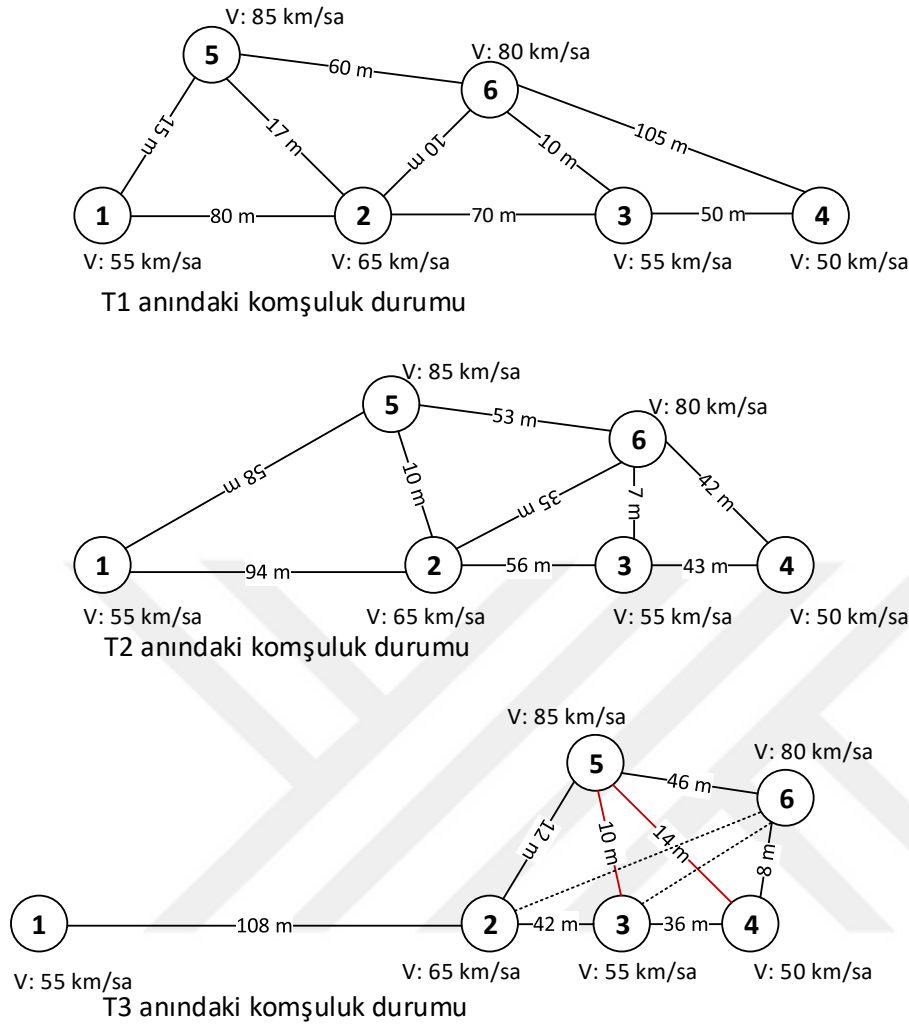
Şekil 3.11. RDP paketlerinin yayılımı

3.3.3. İleri görüşlü komşuluk algoritması

VNPER protokolünde “İleri görüşlü komşuluk” isminde yeni bir yaklaşım öne sürülmüştür. Sorgunun yapıldığı sırada iki düğümün aralarında komşu olma durumu, bir sonraki zaman diliminde komşu olma durumundan daha az önemlidir. Hatta hedef ile kaynak arasındaki mesafe ve araçların birbirlerine göre olan bağlı hızları komşuluk ilişkisini etkilemektedir. Örneğin, birbirlerine göre bağlı hızları 0 km/sa’e yakın olan araçlarda komşuluk daha uzun süre sürdüreceği için, bunlar daha güvenilir komşu olarak kabul edilirken, birbirlerine göre

bağlı hızları 100 km/sa olan araçların komşuluğu önemsiz sayılabilir. Sonuç olarak araçsal ağlar için bir düğümün birkaç zaman dilimi sonrasında başka bir düğüm ile komşuluğu daha önemlidir. Bu noktadan yola çıkılarak, ileri görüşlü komşuluk tespiti yapan bir algoritma geliştirilmiştir. Buna ek olarak, ileri görüşlü komşuluk ilişkisinin ne kadar güvenilir olacağını belirleyen matematiksel bir model formüle edilmiştir. Güvenilirlikten kastedilen, iki komşu düğüm arasında sağlıklı iletişim yapabilecek zaman miktarıdır. Özetle, birkaç zaman sonraki topolojinin önceden tahmin edilebilmesidir. VANET gibi ortamlarda yüksek hızlarda topoloji değişimi tahmin edilebilirse, komşuluk tablosu ve birkaç zaman sonraki ağ topolojisi tespit edilebilir. Komşuluk tablosundan yola çıkarak da düğümlerdeki yönlendirme tablosu oluşturulabilir. Düğümlere gelen keşif isteklerine ise bir kaç zaman sonraki topolojiye göre, hedef düğümünün bilinip bilinmediğine dair bir cevap paketi gönderilebilir. Verilen bu cevabın doğruluk derecesi ise, tahmine dayalı olarak geliştirilen algoritmaya göre hesaplanabilir. Bu cevabı alan kaynak düğüm ise, bu ihtimallerden hangisi daha yüksek ise onu yönlendirme tablosuna, diğerlerini ise topoloji tablosunda yedek rota olarak saklayabilir.

Örnek olarak, Şekil 3.12’de ileri görüşlü komşuluk yaklaşımının topolojisi gösterilmiştir. T1 anında 1 nolu düğüm, 4 nolu düğüm ile veri alışverişi yapmak için keşif paketlerini ağa gönderir. T1 anında 1 nolu düğüm 5 ve 2 nolu düğüm ile halihazırda komşu durumundadır. 1 nolu düğüm, T1 anındaki komşuluk tablosuna bakarak sıçrama noktasını belirleyecek olursa 5 nolu düğümü sıçrama noktası olarak seçer. Ancak, 5 nolu komşu seçilerek oluşturulan yönlendirme tablosu, 2 nolu komşu seçilerek oluşturulan yönlendirme tablosundan daha az güvenilirdir. Çünkü, T2 anına geçildikten kısa bir süre sonra bağlantı koptuğundan dolayı, komşuluk ilişkisi son bulur. Bundan dolayı da, yönlendirme tablosu güncellenene kadar paketler düşürülür. Bağlı hıza göre, birbirlerine hızları yakın olan düğümlerin aralarında kurmuş oldukları komşuluklar daha geç bozulacağı için, daha güvenilir rotalar oluşturulabilir. Öğrenilen rotaların öncelik sıralaması da bu güvenilirlik durumları kıyaslanarak sağlanır. Şekil 3.12’deki topolojide, en güvenilir rota 1 – 2 – 3- 4 rotasıdır. 1 – 2 – 5 – 4 rotasının toplam bağlı hızı daha fazla olduğu için daha az güvenilir rotadır.



Şekil 3.12. Ardışık T zamanlarında komşuluk durumları

Kaynak ve hedef düğüm arasındaki rotaların güvenilirlik oranını tespit etmek için aşağıdaki gibi matematiksel bir model öne sürülmüştür. Güvenilirlik oranı düşük olan rotanın maliyeti daha düşük olduğundan, daha iyi rota olarak değerlendirilmektedir.

RT_k herhangi bir k rotasının güvenilirlik değeri

V_α bir düğümün hızı

V_β bir düğümün komşusunun hızı

V_γ bir düğümün komşusunun komşusunun hızı

Kendisinden, komşusundan ve komşusunun komşusundan oluşan 3'lü düğüm grubu m .

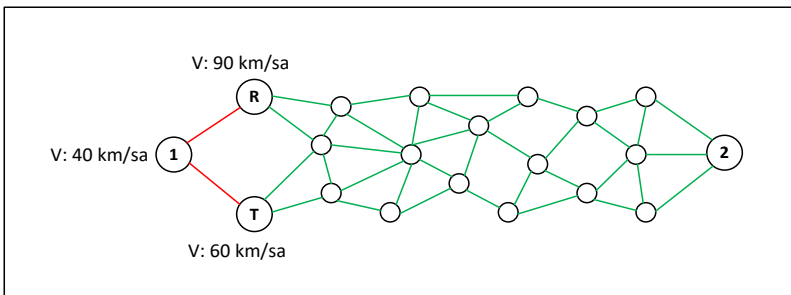
Buna göre, herhangi bir rotanın güvenilirlik değeri Eş.3.1'deki gibi hesaplanır.

$$(RT_k) = \sum_{m=1}^n (|V_\alpha - V_\beta| + |V_\beta - V_\gamma|)_m \quad (3.1)$$

Hesaplamanın yapılabilmesi için aşağıdaki koşulun sağlanması gerekmektedir. Buradaki 40 değeri, komşu iki düğüm arasındaki bağıl hızı ifade etmektedir. Komşu düğümler arasındaki bağıl hız 40 km/saat'den büyük ise, rota üzerindeki diğer düğümlere bakılmaksızın ilgili rota değerlendirmeye alınmamaktadır. Bu sınır komşuluk güven indeksi olarak isimlendirilmiştir. Benzetim ortamında yapılan deneysel sonuçlar neticesinde, 40 km/saat en uygun komşuluk güven indeksi olarak belirlenmiştir. İki düğüm arasındaki hız farkı, Eş. 3.2'deki gibi hesaplanır.

$$|V_\alpha - V_\beta| < 40 \quad (3.2)$$

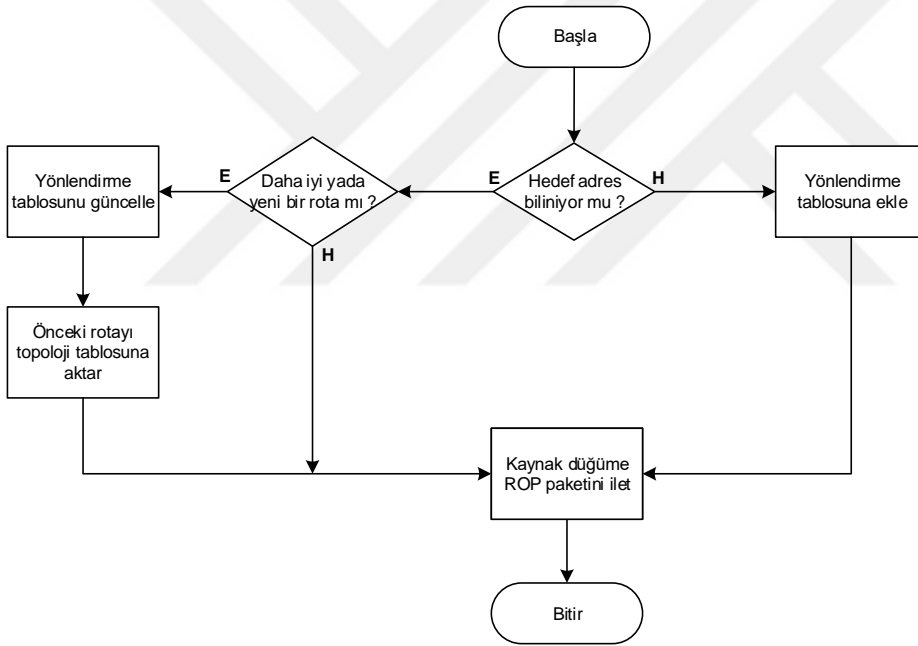
Şekil 3.13'de komşuluk güven indeksinin önemi gösterilmiştir. 1 nolu düğüm, 2 nolu düğüm ile iletişim kuracağı zaman verileri göndermek için tercih edebileceği R ve T isimli 2 adet komşusu bulunmaktadır. R düğümü üzerinden 2 nolu düğümüne doğru belirlenen rotanın güvenilirlik değeri T düğümü üzerinden belirlenen rotadan daha düşük olsun. Komşuluk güven indeksi göz önünde bulunudurulmazsa, veri iletimi için R düğümü sıçrama noktası olarak seçilecektir. Ancak, çok kısa bir süre sonra, R düğümü ile 1 nolu düğüm arasındaki yüksek hız farkından dolayı komşuluk kopacağından iletilen paketler düşürülmeye başlanacaktır. Yönlendirme protokolü yeni bir rota keşfi yapana kadar ya da rota değişikliğine gidene kadar geçen sürede paket kayıpları yaşanacaktır. Bu sorunun üstesinden gelebilmek için, komşuluk güven indeksi olarak isimlendirilen bir hız farkı sınırı belirlenmiştir. Komşuluk güven indeksi sayesinde, 1 nolu düğüm R düğümü üzerinden belirlenen rotanın güvenilirlik değeri daha düşük olmasına rağmen, T düğümünü sıçrama noktası olarak seçmektedir.



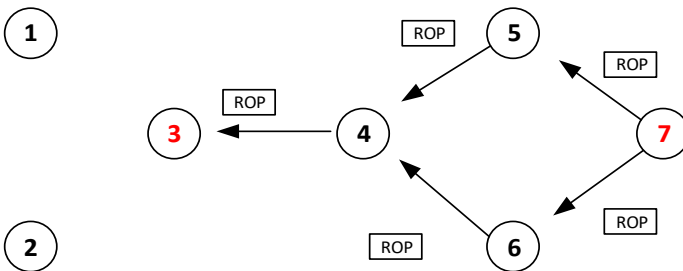
Şekil 3.13. Komşuluk güven indeksi

3.3.4. VNPER protokolünün rota teklif akış şeması

RDP paketini alan bir düğüm, eğer hedefi biliyorsa ya da hedef kendisi ise üreteceği bir paket tipidir. Üretilen bu paketlere ROP (Route Offer Packet) adı verilmiştir. Şekil 3.14’de bu adımın akış şeması gösterilmiştir. ROP paketini alan bir düğüm ilk olarak, paketin kaynak adresinin bilinip bilinmediğini kontrol etmektedir. Bilinmiyorsa yönlendirme tablosuna ekler. Biliniyorsa, aynı kaynak için yönlendirme tablosunda daha iyi bir rotanın olup olmadığı kontrol edilmektedir. Eğer yoksa, ROP paketi hedef düğüme doğru iletilir. Eğer varsa, yönlendirme tablosunda daha iyi rota olacak şekilde güncellenir ve önceki rota kaydı topoloji tablosuna aktarılır. Daha sonra hedef düğüme doğru ROP paketi iletilir. ROP paketlerinin düğümler arasında yayılması Şekil 3.15’de gösterilmiştir.



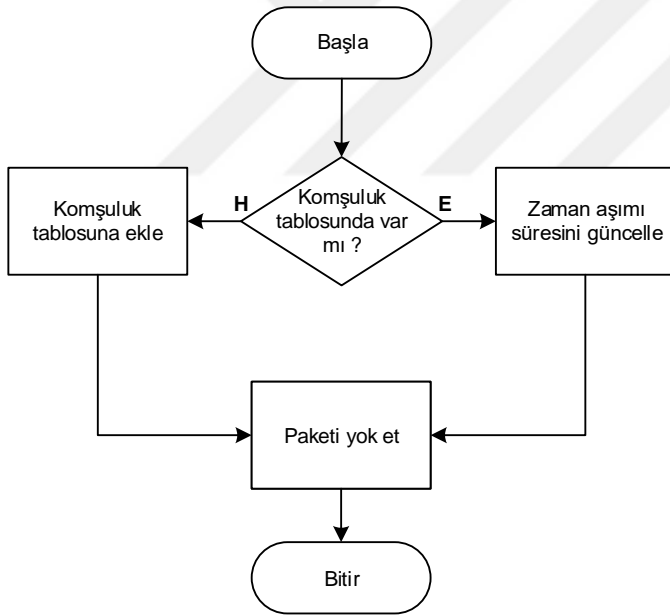
Şekil 3.14. Rota teklif paketinin akış şeması



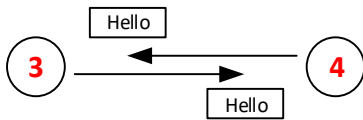
Şekil 3.15. ROP paketlerinin yayılımı

3.3.5. VNPER protokolünün komşu keşif akış şeması

VANET ortamında düğümlerin kendi komşularını bilmesi gerekmektedir. Düğümler sabit olmadıkları için komşuluklar sürekli olarak değişmektedir. Bu dinamiklik sürekli olarak hello paketlerinin değişimi ile takip edilebilir. Şekil 3.16'da komşu düğüm keşfinin akış şeması gösterilmiştir. Bir düğüm hello paketi aldığı zaman, ilk olarak hello paketini gönderen düğümün komşuluk tablosunda olup olmadığını kontrol etmektedir. Eğer komşuluk tablosunda yoksa, yeni bir komşuluk kurulacağını düşünüp, komşuluk tablosuna bu bilgiyi ekler. Eğer komşuluk tablosunda hello paketini gönderen düğüm varsa, komşuluk tablosunda düğümlerin bulunma süresini tazeler. Bu süre içerisinde, komşu düğümden tekrar hello paketi gelmezse komşunun artık olmadığı düşünülüp, tablodan silinir. İlk defa komşuluk kurulması bu şekilde gerçekleştirilmektedir. Hello paketlerinin düğümler arasında yayılması Şekil 3.17'de gösterilmiştir.



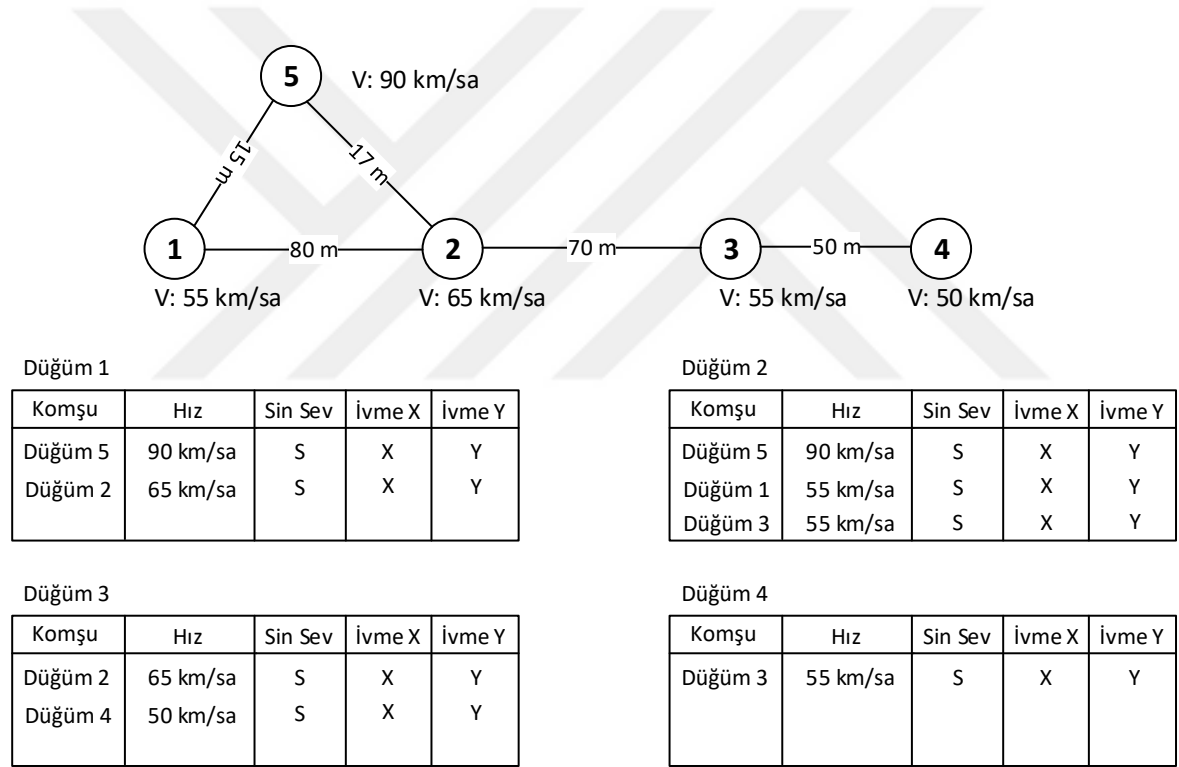
Şekil 3.16. Komşuluk keşif akış şeması



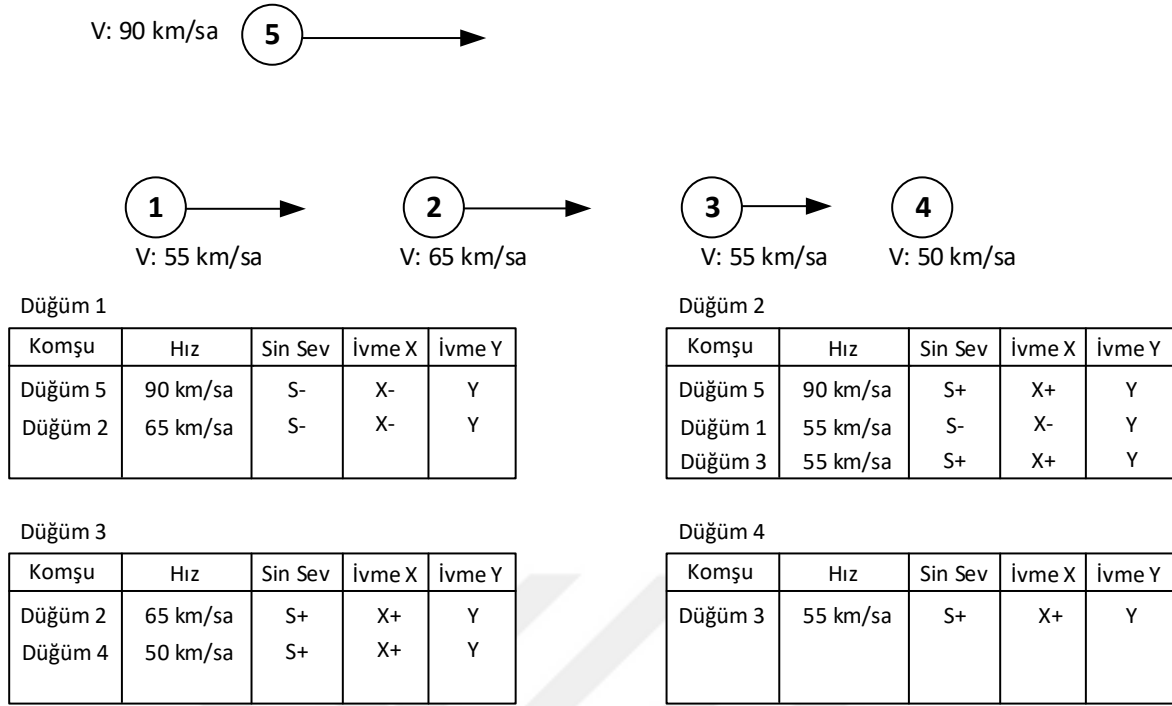
Şekil 3.17. Hello paketlerinin yayılımı

3.3.6. VNPER protokolünün ileri görüşlü komşu keşif işlemleri

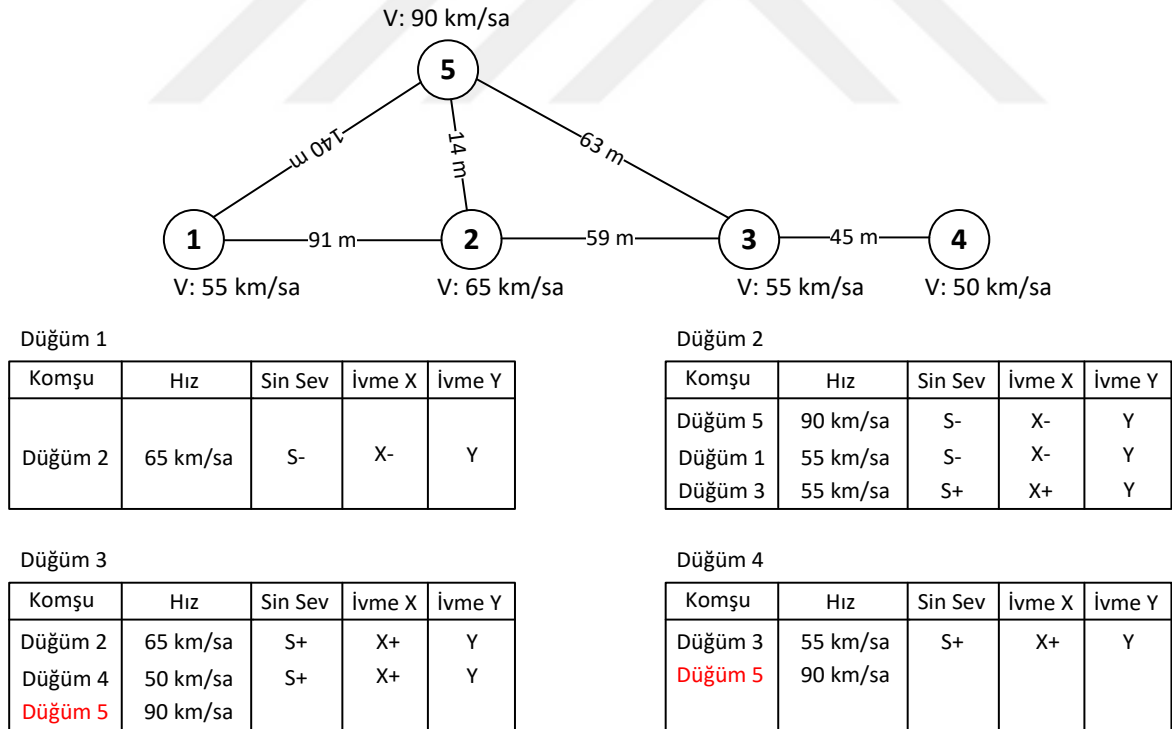
İki düğüm arasında ilk defa komşuluk ilişkisinin kurulması, bir önceki kısımda anlatıldığı gibi gerçekleşir. Ancak, VANET ortamındaki düğümlerin yüksek hızlarda hareket etmesinden dolayı, komşuluklar sürekli olarak değişmektedir. Bundan dolayı, gelecekte bir düğümün hangi düğümlerle komşuluk kuracağını tahmin edebilmesi oldukça önemlidir. VNPER protokolünde kullanılmak için geliştirilen ileri görüşlü komşu keşif yaklaşımı ile gelecekteki komşuluk durum tespiti yapılabilir. İleri görüşlü komşu keşif yaklaşımı Şekil 3.18, 3.19 ve 3.20’de açıklanmıştır. Düğümlerin T1, T1 dinleme ve T2 anındaki durumları sırasıyla Şekil 3.18, 3.19 ve 3.20’de gösterilmiştir.



Şekil 3.18. T1 anında düğümlerin ileri görüşlü komşuluk tablosu



Şekil 3.19. T1 dinleme anında düğümlerin ileri görüşlü komşuluk tablosu



Şekil 3.20. T2 anında düğümlerin ileri görüşlü komşuluk tablosu

Herhangi T1 anında düğümlerin oluşturmuş olduğu topoloji Şekil 3.18'deki gibi olsun. Bu topolojiye göre düğümlerdeki ileri görüşlü komşuluk tablosu Şekil 3.18'deki gibi olur. T2 anındaki düğümlerin komşuluk tablosu, T1 dinleme anının sonunda tespit edilebilmektedir.

Görüldüğü gibi, Şekil 3.18’de T1 anındaki komşuluk tablosunda her bir düğümün komşuları, komşuların hızları, aralarındaki sinyal seviyeleri, koordinat düzleminde X ve Y eksenine göre ivmeleri bulunmaktadır. Bir sonraki zaman dilimi T1 dinleme olarak adlandırılmıştır. T1 dinleme anında görüldüğü gibi, düğümlerin komşuları ile aralarındaki sinyal seviyelerinde, koordinat düzlemine göre X eksen yönündeki ivmesinde değişimler olmaktadır. S- sinyal seviyesinin azaldığı ve dolayısı ile düğümlerin birbirlerinden uzaklaştığı sonucuna ulaşılabilir. S+ sinyal seviyesinin arttığı ve dolayısı ile düğümlerin birbirlerine yaklaştığı sonucuna ulaşılabilir. Buna ek olarak, T1 dinleme anında düğümlerin X ve Y koordinatlarına göre ivme bilgileri birbirlerine gönderilmektedir. X+ koordinat düzleminde X yönünde düğümlerin birbirlerine yaklaştığı, X- ise düğümlerin birbirlerinden uzaklaştığı anlamına gelmektedir. Y+ koordinat düzleminde Y yönünde düğümlerin birbirlerine yaklaştığı, Y- ise düğümlerin birbirlerinden uzaklaştığı anlamına gelmektedir. T1 anında komşuluk tablolarının düğümler arasında değişimi ile komşu düğümlerin hızlarını, T1 dinleme anında ise, komşu düğümün ve bir sıçrama ile ulaşabildiği bir sonraki düğümün kendisine yaklaşıp yaklaşmadığını anlayabilir. Ayrıca, T1 dinleme anında düğümlerin ivme bilgileri elde edilmektedir. Bu sayede, düğümlerin birbirlerine hangi yönden yaklaştığı veya uzaklaştığı tespit edilmektedir. Bu işlemlerin sonunda, 3 nolu düğüm T2 anında 5 nolu düğüm ile komşu olabileceğini anlayabilmektedir. Ayrıca, T3 anında 4 nolu düğüm de 5 nolu düğüm ile komşu olabileceğini tespit edebilmektedir.

Düğümlerin koordinat sistemine göre birbirlerine hangi yönden yaklaşıp azaldığı geliştirilen ivme hesaplama fonksiyonu sayesinde tespit edilmektedir. Şekil 3.21’de koordinat düzlemi üzerinde VANET ortamındaki düğümlerin pozisyonları gösterilmiştir. T1 zamanında Z ve K düğümlerinin X ve Y koordinatları şekildeki gibi olsun. Bu durumda Z ve K düğümlerinin X ve Y yönündeki ivmesi aşağıdaki gibi hesaplanır.

$$\text{İvme } X_Z = \text{Düğümün } x \text{ koordinatı} \times \text{hız} \quad (3.3)$$

$$\text{İvme } Y_Z = \text{Düğümün } y \text{ koordinatı} \times \text{hız} \quad (3.4)$$

$$\text{İvme } X_K = \text{Düğümün } x \text{ koordinatı} \times \text{hız} \quad (3.5)$$

$$\text{İvme } Y_K = \text{Düğümün } y \text{ koordinatı} \times \text{hız} \quad (3.6)$$

Z ve K düğümlerinin ivme farkları kullanılarak birbirlerine göre olan durumları belirlenmektedir. K düğümüne, Z düğümün hangi yönden yaklaştığı ya da uzaklaştığı aşağıdaki gibi hesaplanmaktadır. Pozisyon K_X X koordinatına göre, Pozisyon K_Y Y

koordinatına göre pozisyon bilgisini temsil etmektedir. D ğ mlerin arasındaki mesafenin azalıp azalmadıđı ise, *Pozisyon* K_X ve *Pozisyon* K_Y deđerlerinin mutlak deđeri alınarak hesaplanır ve sırasıyla Eş. 3.9 ve Eş. 3.10 ile g sterilmiřtir.

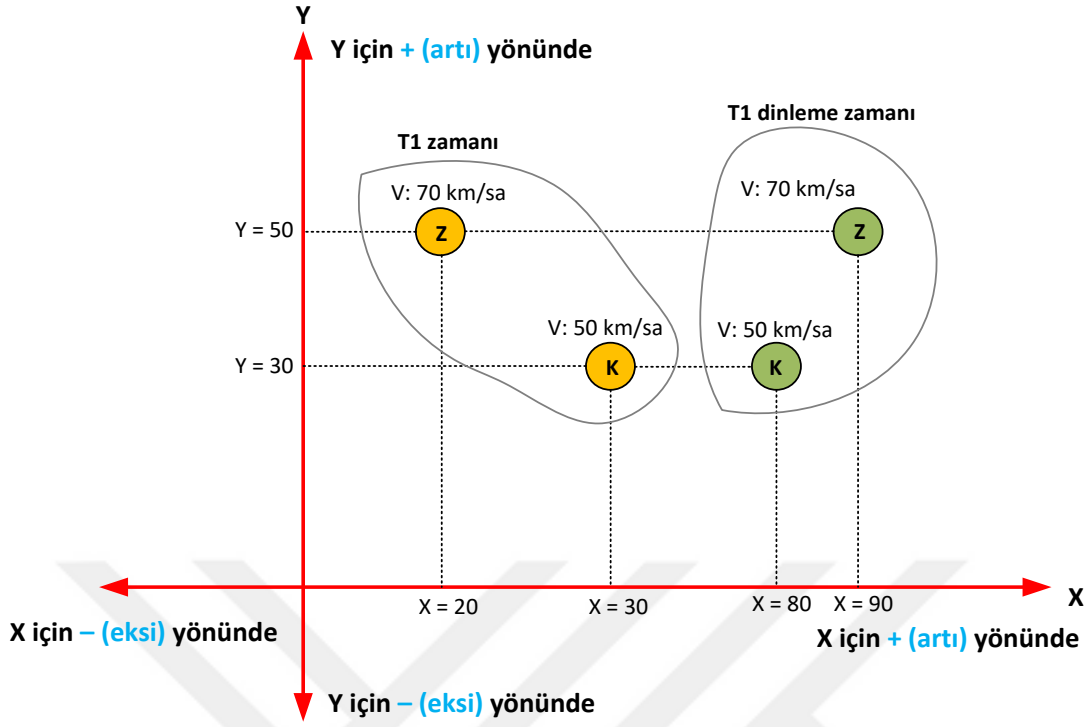
$$\text{Pozisyon } K_X = \text{Ivme } X_K - \text{Ivme } X_Z \quad (3.7)$$

$$\text{Pozisyon } K_Y = \text{Ivme } Y_K - \text{Ivme } Y_Z \quad (3.8)$$

$$\text{Mesafe } K_X = |\text{Pozisyon } K_X| \quad (3.9)$$

$$\text{Mesafe } K_Y = |\text{Pozisyon } K_Y| \quad (3.10)$$

řekil 3.21'de T1 zamanında Z d ğ m n n X koordinat deđeri 20, Y koordinat deđeri 50 ve K d ğ m n n X koordinat deđeri 30, Y koordinat deđeri 30'dur. Bu durumda ivme hesaplama fonksiyonu kullanılarak Z d ğ m n n X y n ndeki ivmesi 1400, Y y n ndeki ivmesi 3500'd r. Benzer řekilde, K d ğ m n n X y n ndeki ivmesi 1500, Y y n ndeki ivmesi 1500 olmaktadır. T1 anı ile T1 dinleme anı arasında geen s re ierisinde d ğ mler sadece X y n nde hızları kadar mesafe ilerlemiř olsun. Bu durumda T1 dinleme anında; Z d ğ m n n X y n ndeki ivmesi 6300, Y y n ndeki ivmesi 3500'd r. Benzer řekilde, K d ğ m n n X y n ndeki ivmesi 4000, Y y n ndeki ivmesi 1500 olmaktadır. Bu durumda, T1 anında X y n ndeki ivmeler dikkate alınarak K d ğ m n n ivmesinden Z d ğ m n n ivmesi ıkartılarak, $\text{Pozisyon } K_X = 100$ sonucuna ulařılır. Benzer řekilde T1 dinleme anındaki $\text{Pozisyon } K_X = -2300$ sonucuna ulařılır. Ardıřık iki zaman diliminde elde edilen bu sonular karřılařtırıldıđında, $\text{Pozisyon } K_X$ sonucunun iřaretinin deđiřmesi, Z d ğ m n n K d ğ m n n , X kordinatına g re ya  n nden arkasına ya da arkasından  n ne getiđi sonucuna ulařılır. Elde edilen sonucun mutlak deđerinin alınması ise aradaki mesafenin artması ya da azalmasının tespiti iin kullanılır. Aradaki fark arttıķa d ğ mlerin birbirlerinden dođru orantılı olarak uzaklařması, fark 0'a yaklařtıķa d ğ mlerin birbirlerine yaklařtıđı anlamına gelmektedir.



Şekil 3.21. Koordinat düzleminde düğümlerin pozisyonları

Benzetim ortamında yapılan düzenleme sayesinde, düğümlerin pozisyon, hız ve ivme bilgilerinin elde edilmesi sağlanmıştır. Benzetim ortamında yapılan değişikliklerin ekran görüntüsü Şekil 3.22, 3.23, 3.24 ve 3.25 de gösterilmiştir.

```

/**This code retrieves node position*****/
iNode = (MobileNode*) (Node::get_node_by_address(index));
((MobileNode *) iNode)->getLoc(&xpos,&ypos,&zpos);
printf("Zaman (%.6f) = %d nolu dugumun X Kordinati : %f, Y Kordinati : %f, Z Kordinati : %f \n", CURRENT_TIME, index, xpos, ypos, zpos);
fprintf(posf,"Zaman (%.6f) = %d nolu dugumun X Kordinati : %f, Y Kordinati : %f, Z Kordinati : %f \n", CURRENT_TIME, index, xpos, ypos, zpos);

```

Şekil 3.22. Düğümlerin pozisyon bilgisinin getirilmesi için eklenen kod parçası

```

/**This code retrieves the nodes velocity*****/
iNode = (MobileNode*) (Node::get_node_by_address(index));
((MobileNode *) iNode)->getVelo(&xpos, &ypos, &zpos);
printf("Zaman (%.6f) = %d nolu dugumun X Kordinati : %f, Y Kordinati : %f, Z Kordinati : %f \n", CURRENT_TIME, index, xpos, ypos, zpos);
fprintf(velf,"Zaman (%.6f) = %d nolu dugumun X Kordinati : %f, Y Kordinati : %f, Z Kordinati : %f \n", CURRENT_TIME, index, xpos, ypos, zpos);

```

Şekil 3.23. Düğümlerin ivme bilgisinin getirilmesi için eklenen kod parçası

```

/**This code retrieves the nodes speed*****/
iNode = (MobileNode*) (Node::get_node_by_address(index));
node_speed = ((MobileNode *) iNode)->speed();
printf("Zaman (%.6f) = %d nolu dugumun hızı = %d \n", CURRENT_TIME, index, node_speed);
fprintf(hizf,"Zaman (%.6f) = %d nolu dugumun hızı = %d \n", CURRENT_TIME, index, node_speed);

```

Şekil 3.24. Düğümlerin hız bilgisinin getirilmesi için eklenen kod parçası

```

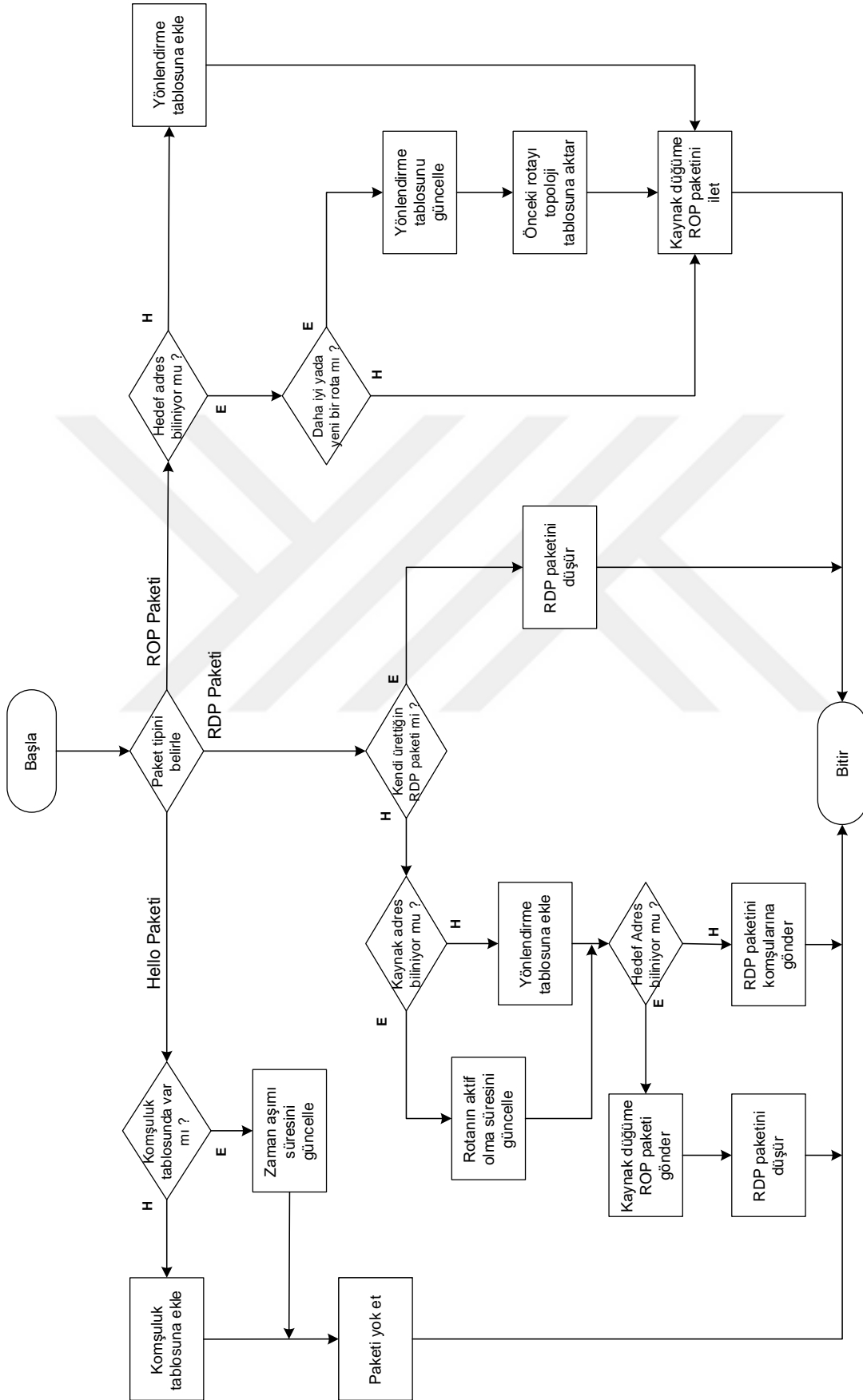
class MobileNode : public Node
{
    friend class PositionHandler;
public:
    MobileNode();
    virtual int command(int argc, const char*const* argv);

    double distance(MobileNode*);
    double propldelay(MobileNode*);
    void start(void);
    inline void getLoc(double *x, double *y, double *z) {
        update_position(); *x = X_; *y = Y_; *z = Z_;
    }
    inline void getVelo(double *dx, double *dy, double *dz) {
        *dx = dX_ * speed_; *dy = dY_ * speed_; *dz = 0.0;
    }
    inline MobileNode* nextnode() { return link_.le_next; }
    inline int base_stn() { return base_stn_; }
}

```

Şekil 3.25. İvme fonksiyonu hesaplaması yapan kod parçası

İleri görüşlü komşuluk yaklaşımı sayesinde, bir sonraki zaman dilimindeki komşuluk durumları tespit edilerek VANET ortamının topolojisi tahmin edilebilmektedir. Bir sonraki zaman dilimindeki komşuluk durumlarına göre yönlendirme tablosu ve topoloji tablosuna gelecekte olası rota kayıtları eklenebilir. Bu sayede, bir sonraki zaman diliminde yeniden komşuluk kurulması ve buna bağlı olarak yeniden rota keşfi yapılması işleminden kaynaklanan paket kayıpları ve gecikmelerin önüne geçilmiş olunur. Özetle, araçların yüksek hızlarından kaynaklanan topolojik yapının ve rotaların değişmesi, önceden tahmin edilerek ağın verimliliği ileri görüşlü komşuluk yaklaşımı sayesinde artırılmaktadır. Şekil 3.26'da VNPER protokolünün algoritmasının akış şeması gösterilmiştir.



Şekil 3.26. VNPER protokolünün akış şeması

4. UYGULAMA ORTAMI VE SENARYO

VANET ortamları için protokol geliřtirmek, test yapmak ve performans ölçümleri gerçekleřtirmek için bazı benzetim ortamları geliřtirilmiřtir. VANETsim, VNS, VanetMobiSim, NS2-Move-Sumo'nun bütünleřik hali bunlardan bazılarıdır. VANET ortamında gerçekçi testlerin yapılabilmesi için gerçek dünya haritalarına ihtiyaç duyulmaktadır. Gerçek dünya haritalarının benzetim ortamına aktarılması için, GDF Map, Tiger Map, Clustered Voronoi graph, OSM gibi araçlar mevcuttur. Literatürde arařtırmacılar ihtiyaca göre ilgili benzetim ortamlarını kullanarak çeřitli performans tesleri gerçekleřtirebilir.

4.1. Kullanılan Benzetim Ortamı

Bu tez kapsamında, protokol geliřtirmek için NS2 benzetim ortamı kullanılmıřtır. Hareketlilik modeli ve trafiğin modellenmesi için Sumo ve Move kullanılmıřtır. Gerçek dünya haritalarının benzetim ortamına aktarılması için OSM kullanılmıřtır. Kullanılan tüm benzetim araçları birbirleri ile birleřtirilerek çalışabilir hale getirilmiřtir. Bu sayede, NS2-Move-Sumo'nun tümleřik yapısı kullanılmıřtır. Gerçek ortam harita entegrasyonu ise, OSM (Open Street Maps) aracı kullanılmıřtır. Ayrıca, linux üzerinde netconverter paketi ile harita çevirme iřlemi gerçekleřtirilmiřtir.

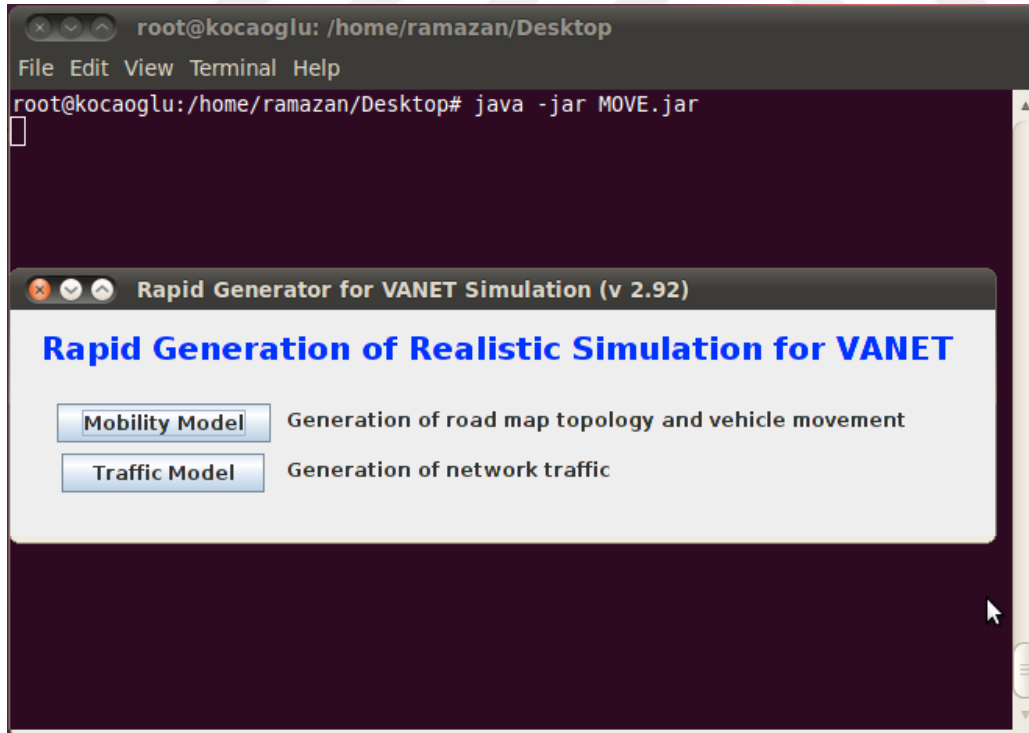
4.1.1. NS2 (Network Simulator 2)

NS2 ilk olarak Berkeley Laboratuvarı'nda geliřtirilmiř çeřitli ağ ortamlarına simüle edebilen ayrıık olay tabanlı bir ağ benzetim aracıdır. Standart hale gelmiř bir çok kuyruk algoritmasını halihazırda içinde barındırmaktadır. NS2 ile kablolu ve kablosuz ağ ortamları simüle edilebilir. Tařıma katmanı protkolü olarak hem TCP hem de UDP protkolü kullanılabilir. Uygulama katmanı olarak günümüzde kullanılan birçok protkole destek vermektedir. Sunmuř olduđu esnek yapısı ve açık kaynak kodlu mimarisi sayesinde, MAC katmanından uygulama katmanına kadar tüm OSI katmanları için protokol tasarlamak mümkündür. NS2, C++ tabanlı bir benzetim ortamı olup, OTCL dili ile bütünleřik olarak çalışarak kullanıcıya bir arayüz sunmaktadır. C++ ile ağdaki her paket için hızlı bir řekilde kontrol iřlemi sağlanırken, OTCL ile mevcutta C++ ile oluřturulan nesnelere kullanarak benzetim

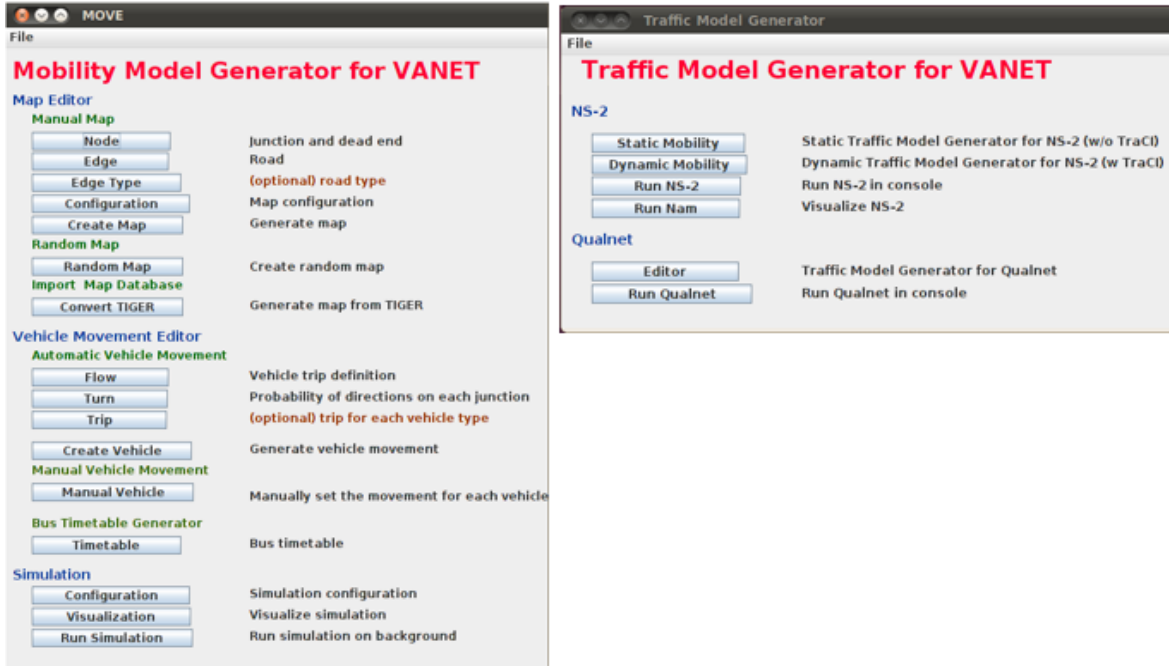
ortamında kullanılacak tüm ayarlamalar yapılmaktadır. Açık kaynak kodlu mimarisinden dolayı esnek bir yapıya sahip olması, literatürde kabul görmesi ve yaygın şekilde kullanılması, araçsal ağlara özgü senaryoları sorunsuz şekilde çalıştırabilmesi gibi sebeplerden dolayı, protokol geliştirmek için benzetim ortamı olarak NS2 kullanılmıştır.

4.1.2. MOVE (MOBility model generator for VEhicular networks)

MOVE benzetim ortamında kullanılacak olan hareketlilik modelini üreten ve NS2 ile tümleşik çalışabilen bir benzetim aracıdır. Gerçek dünya ortamına benzer gerçekçi hareketlilik modelleri oluşturabilen açık kaynak kodlu bir araçtır. MOVE sayesinde, hem otoyollarda hem de şehir merkezlerindeki ortamların gerçek dünya ortamlarına yakın hareketlilik modelleri, araç hızları ve araç sayıları ile modellenmesi mümkündür. Java dili kullanılarak yazılmış bir araçtır. Bu yüzden platform bağımsız çalışabilmektedir. Şekil 4.1'de linux platformunda MOVE'un çalıştırılması, Şekil 4.2'de ise MOVE'un kullanılan arayüzleri gösterilmiştir

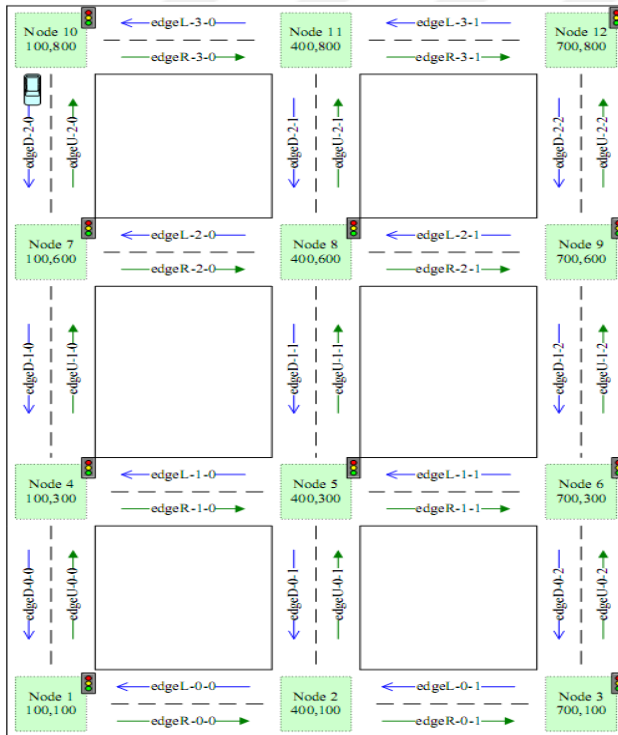


Şekil 4.1. Linux platformunda MOVE'un çalıştırılması



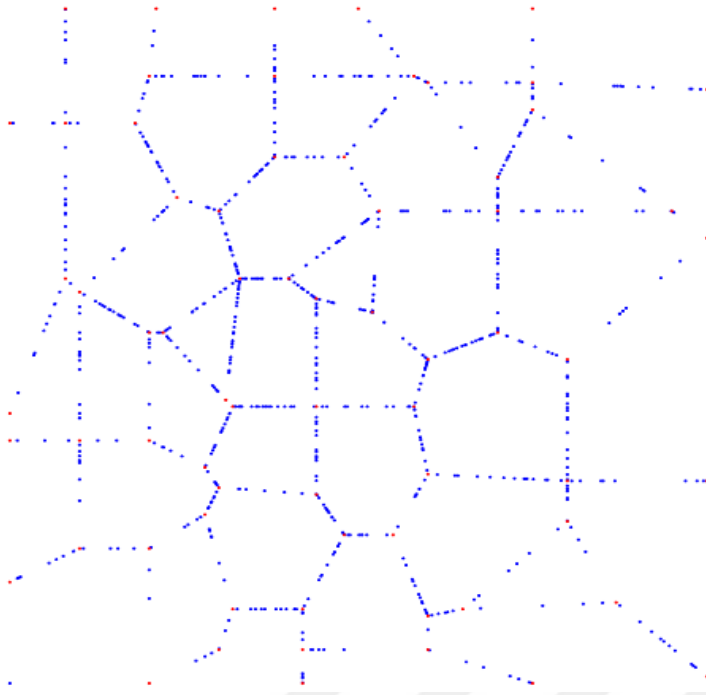
Şekil 4.2. MOVE'un kullanım arayüzü

MOVE aracını kullanarak Şekil 4.3'deki gibi bir benzetim ortamı tasarlanabilir.



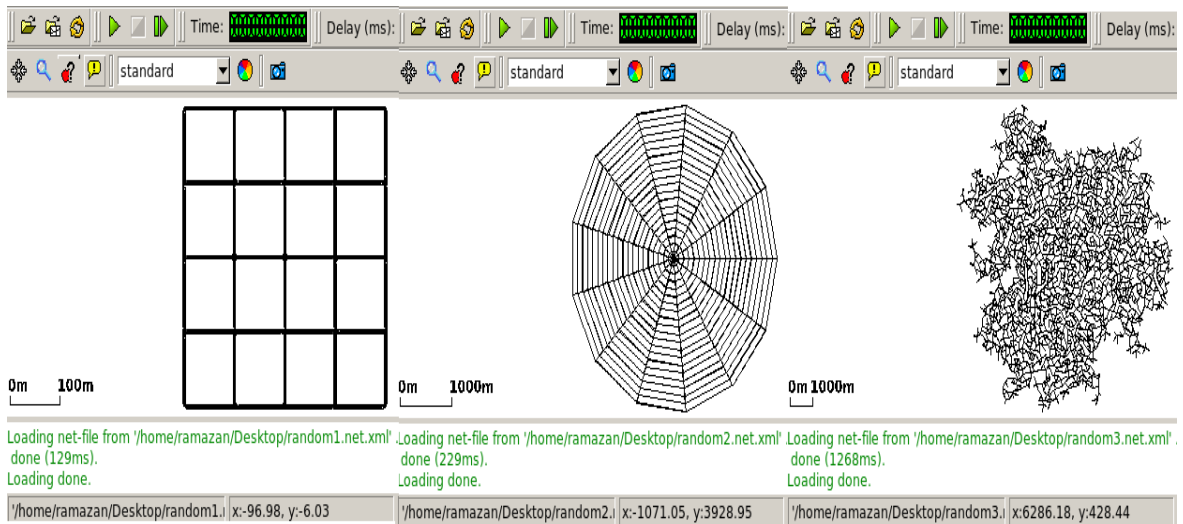
Şekil 4.3. MOVE kullanılarak oluşturulan bir topoloji

Şekil 4.4’de MOVE kullanılarak oluşturulan daha geniş bir topoloji gösterilmiştir.



Şekil 4.4. MOVE kullanılarak hazırlanan geniş bir topoloji

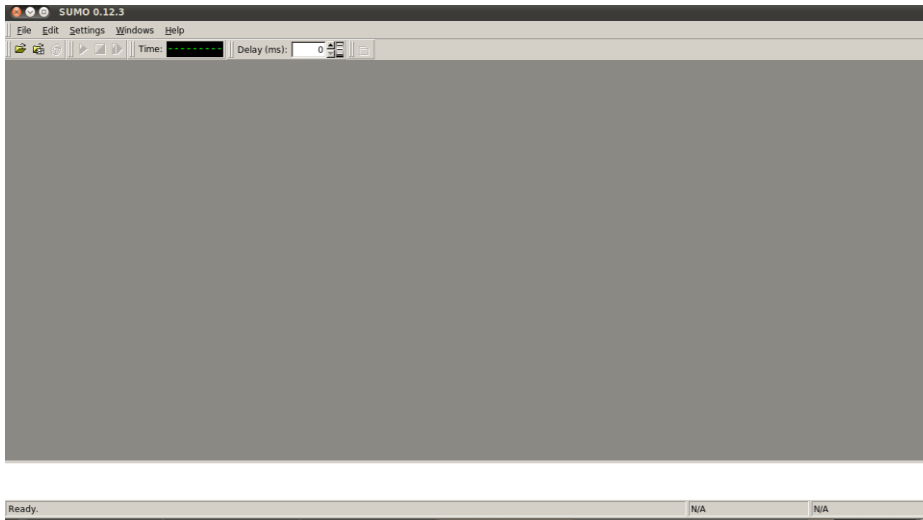
MOVE kullanılarak halihazırda oluşturulabilen bazı topoloji türleri Şekil 4.5’de gösterilmiştir.



Şekil 4.5. MOVE’da halihazırda kullanılabilir topolojiler

4.1.3. SUMO (Simulation of Urban MObility)

SUMO, MOVE ile oluşturulan modellerin NS2 altında sorunsuz şekilde çalışmasını sağlayan bir araçtır. Ayrıca, SUMO ile çok büyük boyutlarda trafik modelleri oluşturulabilir. Modeller büyüdükçe ilgili parametrelerin ayarlanması zorlaşmaktadır. MOVE ile görsel arayüz kullanılarak bu karmaşıklık ortadan kaldırılabilir. SUMO ile gerçek dünya haritaları üzerinde, çalışma yapma imkanı bulunmaktadır. Araçların yollar üzerindeki hareketleri, hızları, trafik yoğunlukları gibi gerçek dünya durumları modellenmektedir. Şekil 4.6’da SUMO’nun genel bir görüntüsü verilmiştir.

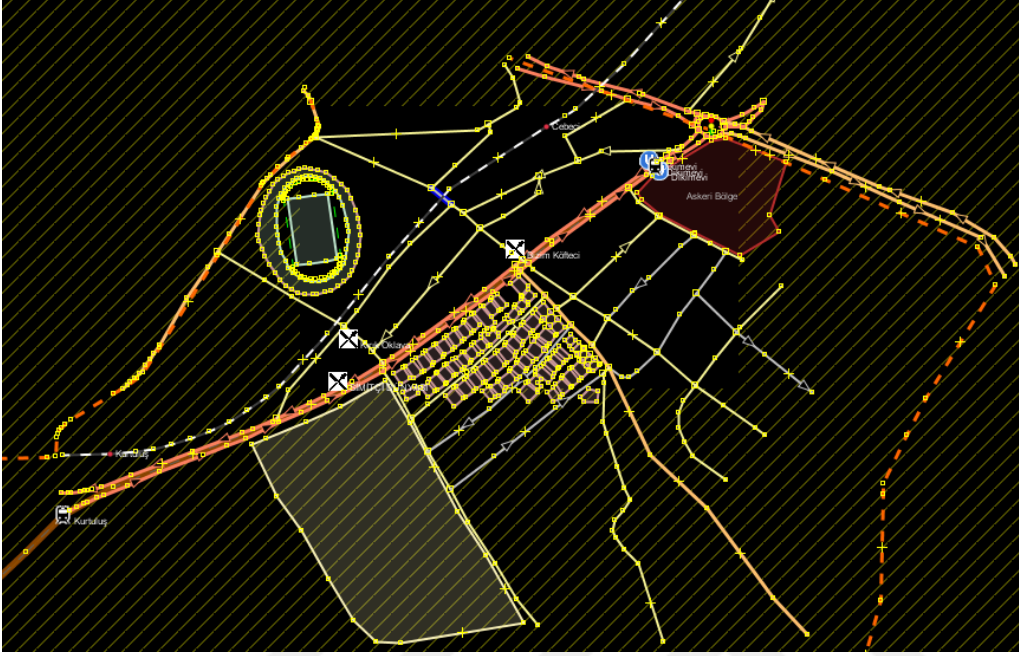


Şekil 4.6. SUMO’nun genel görüntüsü

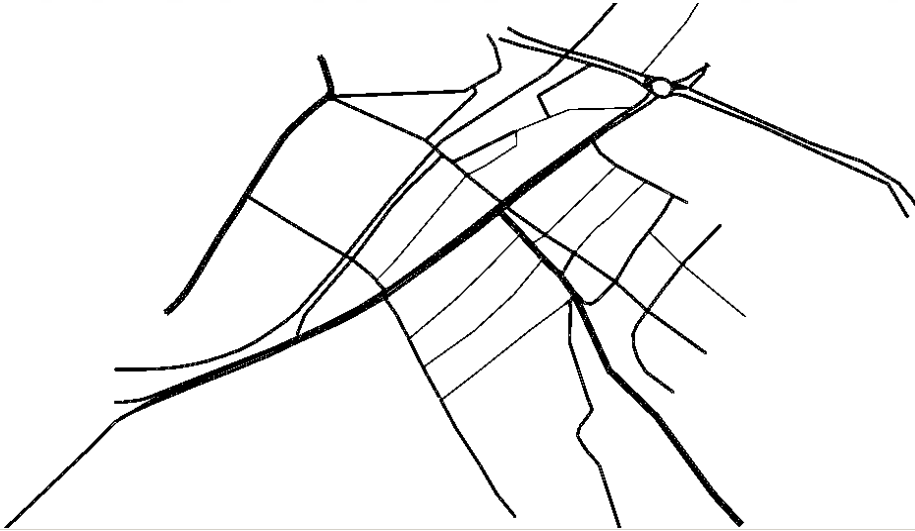
4.1.4. OSM (Open Street Map)

Open Street Map gönüllü katılımcıların destek vererek oluşturdukları dünya haritasını ortaya çıkartma projesidir. Açık kaynak kodlu yazılımlara benzer şekilde, gönüllülük esasına dayanarak katılımcılardan elde edilen GPS izleri, uydu fotoğrafları gibi verilerin bir platform üzerinde birleştirilmesi ile oluşturulmaktadır. Dünya üzerinde çoğu yerel yönetim tarafından da çeşitli veriler sunularak desteklenmektedir. Haritalarda meydana gelen değişikliklerin doğruluğu kontrol edildikten sonra, anında yayınlanmaktadır. Dünya’nın farklı yerlerindeki birçok özel şirket tarafından da desteklenmektedir. Dünya üzerindeki birçok şehrin haritasını halihazırda barındırmaktadır ve bu haritalar geliştirilmeye de devam edilmektedir. Şekil 4.7’de Dikimevi, Ankara Bölgesinin OSM ile elde edilmiş hali gösterilmiştir. Şekil 4.8’de ise, OSM ile elde edilen haritanın netconverter paketi ile SUMO ortamına aktarılacak hali

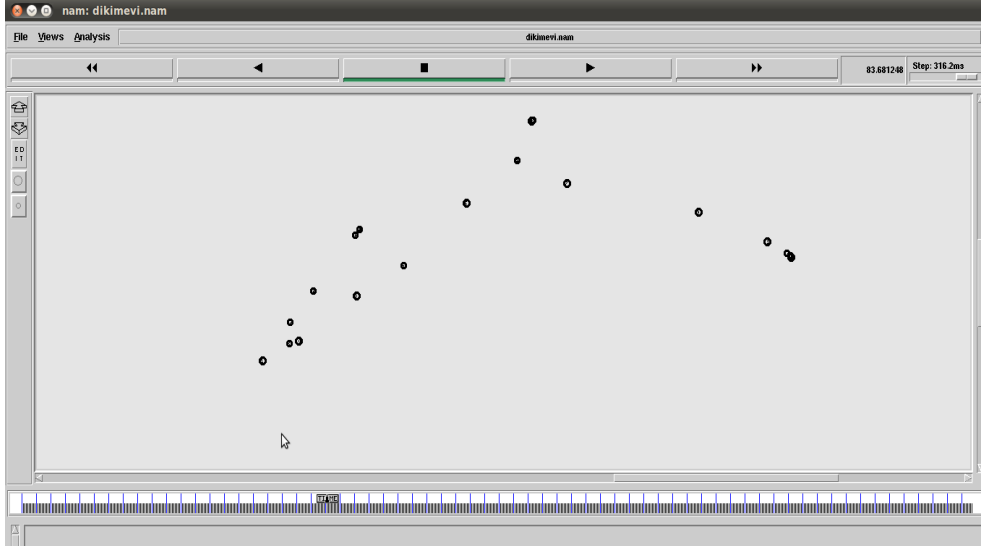
gösterilmiştir. Şekil 4.9’da SUMO üzerinden çalıştırılan haritanın NS2 platformunda çalıştırılmış hali gösterilmiştir.



Şekil 4.7. OSM’den alınan Dikimevi bölgesinin haritası



Şekil 4.8. SUMO için kullanılabilir Dikimevi bölgesinin haritası



Şekil 4.9. Dikimevi bölgesinin NS2 platformunda çalıştırılması

4.2. Uygulama Ortamı ve Haritası

Yönlendirme protokollerinin verimliliğini ortaya çıkarabilmek için, gerçek dünya ortamları üzerinde test edilmesi gerekmektedir. Bu tez çalışmasında, VNPER protokolünün verimliliğini test etmek için kullanılan test haritasının gerçek dünya ortamındaki görüntüsü Şekil 4.10'da gösterilmiştir. Türkiye, Ankara ili içerisinde bulunan Kızılay – Sıhhiye meydanı, Maltepe, Tandoğan, Kolej semtlerini de içine alacak büyük bir alan test ortamı olarak belirlenmiştir. OSM ile bu bölgenin gerçek dünya haritası çıkartılmıştır. Netconverter ile SUMO'da kullanılacak hale getirilmiştir. Gerçek dünya haritasının SUMO ortamına aktarılmış hali Şekil 4.11'de gösterilmiştir. Şekil 4.12'de test haritasında kullanılacak olan Kızılay meydanının bir görüntüsü verilmiştir. Gerçek dünya ortamında olduğu gibi, trafik lambaları kullanılarak ortam benzetimi gerçeğe yakın olacak şekilde sağlanmıştır.



Şekil 4.10. Test ortamı için kullanılan gerçek dünya haritası (Kızılay-Sıhhiye)



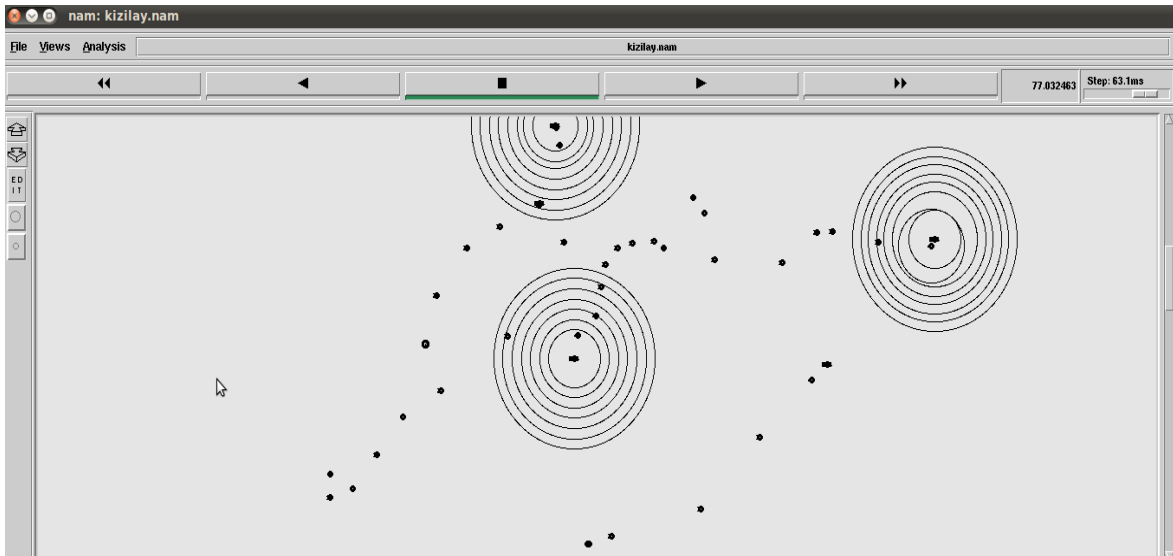
Şekil 4.11. Test ortamı için kullanılan haritanın SUMO görüntüsü (Kızılay-Sıhhiye)



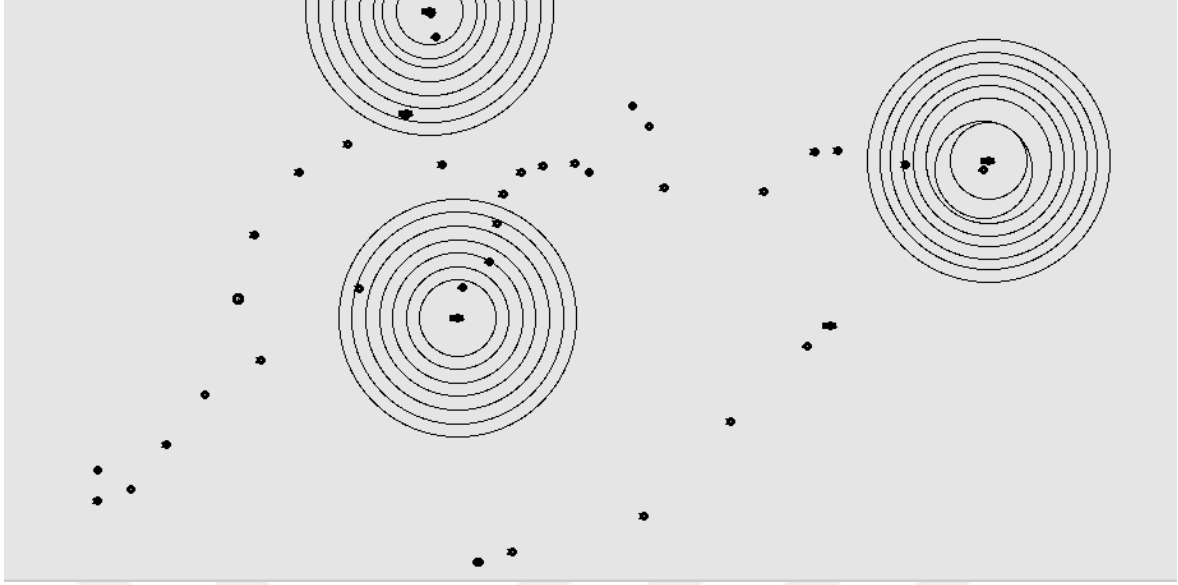
Şekil 4.12. Test ortamından bir görüntü (Kızılay Meydanı)

4.2.1. Uygulama ortamının NS2 ortamına aktarılması

Sumo'da oluşturulan test haritasının parametreleri ve hareket şekilleri NS2 ortamına aktarılmıştır. Bu sayede NS2'nun sunduğu tüm olanaklardan yararlanabilme imkanı olmuştur. Şekil 4.13 ve Şekil 4.14'de NS2 NAM ortamından alınan ekran görüntüleri gösterilmiştir.



Şekil 4.13. Kızılay – Sıhhiye meydanı araç hareketlerinin NS2 ortamındaki görüntüsü



Şekil 4.14. Kızılay – Sıhhiye meydanı araç hareketlerinin NS2 ortamındaki yakınlaştırılmış görüntüsü

4.3. Performans Ölçümü için Kullanılan Metrikler

VNPER protokolünün verimliliğini tespit etmek için bazı metrikler kullanılmıştır. Bu metrikler sayesinde protokolün başarısı diğer protokoller ile kıyaslanmıştır. Kullanılan metrikler; Paket teslim etme oranı, Uçtan uca ortalama gecikme, Birim zamanda gönderilen veri miktarı, Yönlendirme yükü şeklindedir.

4.3.1. Paket teslim etme oranı (PDR)

Literatürde PDR (Packet Delivery Ratio) olarak adlandırılmaktadır. Ağ üzerindeki kaynak düğümden hedef düğüme doğru başarılı şekilde gönderilen paket miktarı oranıdır. Awk dili kullanılarak paket teslim etme oranını ortaya çıkartacak betik yazılmıştır. Ns2'nun senaryo boyunca üretmiş olduğu olaylar tr uzantılı bir dosyaya yazılmaktadır. Bu dosyaya hazırlanan betik uygulanarak PDR oranı tespit edilmektedir. Şekil 4.15'de PDR için yazılan betik gösterilmektedir.

```

BEGIN {
    sendLine = 0;
    recvLine = 0;
    fowardLine = 0;
}

$0 ~/^s.* AGT/ {
    sendLine ++ ;
}

$0 ~/^r.* AGT/ {
    recvLine ++ ;
}

$0 ~/^f.* RTR/ {
    fowardLine ++ ;
}

END {
    printf("\n");
    printf("*****");
    printf("\n");
    printf("cbr s:%d r:%d, r/s Ratio:%.4f, f:%d \n", sendLine, recvLine, (recvLine/sendLine), fowardLine);
    printf("*****");
    print(" ");
    printf("\n");
}

```

Şekil 4.15. Paket teslim etme oranı betiği

4.3.2. Uçtan uca ortalama gecikme (E2E)

Literatürde E2E Delay olarak adlandırılmaktadır. Ağ üzerinde bir paketin kaynağından çıkıp hedefine varması için geçen süreye uçtan uca gecikme denilmektedir. Ağda üretilen tüm paketlerin hedeflerine ulaşması için geçen zamanın ortalaması ise uçtan uca ortalama gecikme olarak adlandırılmaktadır. Awk dili kullanılarak uçtan uca ortalama gecikme miktarını ortaya çıkartacak betik yazılmıştır. Ns2'nun senaryo boyunca üretmiş olduğu olaylar tr uzantılı bir dosyaya yazılmaktadır. Bu dosyaya hazırlanan betik uygulanarak E2E oranı tespit edilmektedir. Şekil 4.16'da E2E için yazılan betik gösterilmektedir. Betik milisaniye türünde sonuç vermektedir.

```

# => Average End-to-End Delay.
# =====
BEGIN {
    seqno = -1;
    droppedPackets = 0;
    receivedPackets = 0;
    count = 0;
    n_to_n_delay=0;
    arg_delay=0;
}
{
    if($4 == "AGT" && $1 == "s" && seqno < $6) {
        seqno = $6;
    }
    else if(($4 == "AGT") && ($1 == "r")) {
        receivedPackets++;
    }
    else if (($1 == "D") && ($7 == "tcp" || $7 == "cbr") && ($8 > 512)){
        droppedPackets++;
    }
}

#end-to-end delay
if($4 == "AGT" && $1 == "s") {
    start_time[$6] = $2;
} else if(($7 == "cbr" || $7 == "tcp") && ($1 == "r")) {
    end_time[$6] = $2;
} else if(($1 == "D") && ($7 == "tcp" || $7 == "cbr")) {
    end_time[$6] = -1;
}
END {
    for(i=0; i<=seqno; i++) {
        if(end_time[i] > 0) {
            delay[i] = end_time[i] - start_time[i];
            count++;
        }
        else
        {
            delay[i] = -1;
        }
    }

    for(i=0; i<=seqno; i++) {
        if(delay[i] > 0) {
            n_to_n_delay = n_to_n_delay + delay[i];
        }
    }

    arg_delay = n_to_n_delay/count;

    print "*****";
    print " ";
    print "GeneratedPackets          = " seqno+1;
    print "ReceivedPackets              = " receivedPackets;
    print "Packet Delivery Ratio        = " receivedPackets/(seqno+1)*100"%";
    print "Total Dropped Packets = " droppedPackets;
    print "Average End-to-End Delay     = " arg_delay * 1000 " ms";
    print " ";
    print "*****";
}

```

Şekil 4.16. Uçtan uca ortalama gecikme betiği

4.3.3. Birim zamanda gönderilen veri miktarı (Throughput)

Literatürde throughput olarak adlandırılmaktadır. Ağ üzerinde belirli bir zaman diliminde tüm düğümlerden alınan toplam veri miktarıdır. Sonuçlar kilobit/saniye olarak sunulmaktadır. Awk dili kullanılarak throughput miktarını ortaya çıkartacak betik yazılmıştır. Ns2'nun senaryo boyunca üretmiş olduğu olaylar tr uzantılı bir dosyaya yazılmaktadır. Bu dosyaya hazırlanan betik uygulanarak throughput oranı tespit edilmektedir. Şekil 4.17'de throughput için yazılan betik gösterilmektedir.

```

BEGIN {
    recvdSize = 0
    startTime = 400
    stopTime = 0
}

{
    event = $1
    time = $2
    node_id = $3
    pkt_size = $8
    level = $4

    # Store start time
    if (level == "AGT" && event == "s" && pkt_size >= 50) {
        if (time < startTime) {
            startTime = time
        }
    }

    # Update total received packets' size and store packets arrival time
    if (level == "AGT" && event == "r" && pkt_size >= 50) {
        if (time > stopTime) {
            stopTime = time
        }
    }

    # Rip off the header
    hdr_size = pkt_size % 512
    hdr_size = 20;
    pkt_size -= hdr_size
    # Store received packet's size
    recvdSize += pkt_size
}

END {
    printf("\n");
    printf("*****");
    printf("\n");
    printf("Average Throughput[kbps] = %.2f\t\t StartTime=%.2f\t\t StopTime=%.2f\n", (recvdSize/(stopTime-startTime)) * (8/1000), startTime, stopTime);
    printf("*****");
    printf(" ");
    printf("\n");
}

```

Şekil 4.17. Toplam veri işleme hacmi

4.3.4. Yönlendirme yükü

Literatürde routing overhead ya da normalized routing load olarak adlandırılmaktadır. Ağ üzerinde kaynaktan hedefe doğru bir paketin iletilmesi için yönlendirme protokolünün üretmiş olduğu paket sayısını ifade etmektedir. Üretilen toplam yönlendirme paketi sayısının hedefe başarılı şekilde iletilen veri paketi sayısına bölünmesi ile elde edilir. Awk dili kullanılarak yönlendirme yükünü ortaya çıkartacak betik yazılmıştır. Ns2'nun senaryo boyunca üretmiş olduğu olaylar tr uzantılı bir dosyaya yazılmaktadır. Bu dosyaya hazırlanan betik uygulanarak yönlendirme yükü miktarı tespit edilmektedir. Şekil 4.18'de yönlendirme yükü için yazılan betik gösterilmektedir.

```
#####
#           AWK Script to calculate Normalized Routing Load           #
#           Works with VNPER, BMFR, CAR, MURU and ASTAR           #
#####

BEGIN{
recvdata = 0;##### to calculate total number of data packets received
routing_pkts = 0;##### to calculate total number of routing packets received
}

{
#### Check if it is a data packet
if (( $1 == "r" ) && ( $7 == "cbr" || $7 == "tcp" ) && ( $4=="AGT" )) recvdata++;

#### Check if it is a routing packet
if (( $1 == "s" || $1 == "f" ) && $4 == "RTR" && ( $7 == "VNPER" || $7 == "BMFR" || $7 == "CAR" || $7 == "MURU" || $7 == "ASTAR" )) routing_pkts++;
}

END{
printf("#####\n");
printf("\n");
printf("                Normalized Routing Load = %.6f\n", routing_pkts/recvdata);
printf("\n");
printf("#####\n");
}
}
```

Şekil 4.18. Yönlendirme yükü scripti

4.4. Uygulama Sonuçları ve Değerlendirme

Uygulama haritası üzerinde VNPER ve diğer yönlendirme protokolleri aynı şartlar altında çalıştırılmıştır ve elde edilen sonuçlar paket teslim etme oranı, uçtan uca gecikme, birim zamanda gönderilen veri miktarı ve yönlendirme yükü metriklerine göre kıyaslanmıştır. Aynı test ortamı üzerinde 4 farklı benzetim model kullanılarak deneyler gerçekleştirilmiştir. İlk modelde, veri trafik miktarı sabit tutulup araç sayısı artırılarak sonuçlar alınmıştır. İkinci modelde araç sayısı sabit tutulup veri trafik miktarı artırılarak sonuçlar alınmıştır. Üçüncü modelde, hem araç sayısı hem de veri trafik miktarı artırılarak sonuçlar alınmıştır. Dördüncü modelde ise, araç sayısı ve trafik miktarı sabit tutularak veri iletişimde bulunacak olan düğüm çiftlerinin sayısı artırılmıştır. VNPER protokolünün performansını ortaya çıkarabilmek için her modelde elde edilen sonuçlar diğer araçsal yönlendirme protokolleri olan STAR, MURU, CAR, BMFR protokolleri ile kıyaslanarak uygulama sonuçları elde edilmiştir. VNPER protokolü, herbiri farklı bir çözüm algoritmasına sahip olduğu için bu protokoller ile kıyaslanmıştır. Deneysel sonuçlar, dört benzetim modelinde de VNPER protokolünün diğer yönlendirme protokollerine oranla daha başarılı olduğunu göstermiştir.

4.4.1. Birinci modelden elde edilen deneysel sonuçlar

Birinci benzetim modelinde, üretilen veri trafik miktarı sabit tutularak araç sayısı artırılmıştır. Benzetim süresi boyunca toplamda yaklaşık 3000 adet veri paketi üretilmiştir. Farklı sayıda araç farklı noktalardan başlayarak farklı yönlere doğru hareket edecek şekilde 300 saniye boyunca araç trafiği oluşturulmuştur. Araçlar arasında ise veri trafiği oluşturularak kaynak araçtan hedef araca doğru veri iletilmesi sağlanmıştır. Uygulama

ortamında kullanılan NS2 parametreleri Çizelge 4.1’de gösterilmiştir. Oluşturulan veri trafiğinin hedef düğüme ne oranda ulaştığı, ne kadar sürede ulaştığı gibi ölçümler yapılarak protokollerin verimliliği test edilmiştir.

Çizelge 4.1. Birinci modelde kullanılan parametreler

Üretilen Veri Paketi Sayısı	3010 paket
Kanal Tipi	WirelessChannel
Yayılm Modeli	TwoRayGround
Interface Tipi	WirelessPhy
MAC Tipi	802_11
Kuyruk Tipi	DropTail/PriQueue
Anten tipi	OmniAntenna
Bağlantı Katmanı Tipi	LL
Kuyruk uzunluğu	50 paket
X Koordinat uzunluğu	16559 birim
Y Koordinat uzunluğu	6656 birim
Benzetim süresi	300 sn

Uygulama haritası üzerinde bulunan 6 tane farklı noktadan farklı hedeflere doğru araç trafiği başlatılmıştır. Her bir senaryo için, 0. saniyeden 300. saniyeye kadar ortalama 3010 adet veri paketi üretilmiştir. Araç sayılarına göre 5 farklı senaryo oluşturulmuştur. Senaryo 1’de uygulama ortam üzerindeki araç sayıları ve trafik hareketleri Çizelge 4.2’de gösterilmiştir. Senaryo 1 için araç trafiğinin Sumo ortamında oluşturulması ise Şekil 4.19’da gösterilmiştir.

Çizelge 4.2. Birinci model birinci senaryoda üretilen araç sayısı ve trafik akışı

Başlangıç Noktası	Bitiş Noktası	Başlangıç Süresi	Bitiş Süresi	Araç Sayısı
Kurtuluş Kavşağı	G.M.K bulvarı	0	300	15
Kurtuluş Kavşağı	Altınsoy Cad.	0	300	15
Atatürk Bulvarı	Kurtuluş Kavşağı	0	300	20
Strazburg Cad.	Necatibey Cad.	0	300	20
Atatürk Bulvarı	Fevzi Çakmak Sok.	0	300	15
Atatürk Bulvarı	G.M.K bulvarı	0	300	15

```

</flows>
<flow id="flow1" from="12195038#2" to="68695137#5" begin="0" end="300" no="15" />
<flow id="flow2" from="12195038#0" to="28470416#6" begin="0" end="300" no="15" />
<flow id="flow3" from="68695134#3" to="12195038#0" begin="0" end="300" no="20" />
<flow id="flow4" from="26302861#0" to="26317869#4" begin="0" end="300" no="20" />
<flow id="flow5" from="267158069#0" to="46039706#2" begin="0" end="300" no="15" />
<flow id="flow6" from="267158069#1" to="68695130#4" begin="0" end="300" no="15" />
</flows>

```

Şekil 4.19. Birinci model birinci senaryoda araç trafiğinin sumo ortamında oluşturulması

İlk senaryoda, araç miktarı 100 olarak belirlenmiştir ve senaryo 1 olarak isimlendirilmiştir. Araçların sayısı ve trafik akışı tüm protokoller için aynı tutulmuştur. Eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.3’de sunulmuştur. Protokoller, aynı senaryo üzerinde çalıştırıldığı için aynı sayıda veri paketi ve araç üretilmiştir. Üretilen 3010 adet veri paketinden 1497 adeti VNPER protokolü ile hedefine başarılı şekilde ulaştırılmıştır. Paket teslim etme oranı, uçtan uca gecikme ve throughput değeri en başarılı protokol VNPER olmuştur. Aynı metrikler göz önünde bulundurulduğunda, VNPER protokolünden sonra en verimli sonuçları BMFR protokolü elde etmiştir. ASTAR protokolü, diğer protokollere oranla daha düşük performans sonuçları elde etmiştir. Ancak, yönlendirme yükü en az olan protokol ASTAR olurken, en fazla yönlendirme yüküne sahip olan protokol VNPER olmuştur. VNPER protokolü bu açıdan bakıldığında yüksek oranda PDR ve kabul edilebilir oranda E2E ölçüleri ile başarılı şekilde veri iletimini gerçekleştirmiştir. Başarılı şekilde veri iletimini gerçekleştirmek için diğer protokollere oranla daha fazla yönlendirme paketi üretmiştir. Senaryo 1’de VNPER protokolü yaklaşık olarak 18682 adet yönlendirme paketi üretirken, ASTAR protokolü 9479 adet yönlendirme paketi üretmiştir.

Çizelge 4.3. Birinci model birinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Araç Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yükü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	100	1497	49,23	302,33	12,48	18682	34,87
ASTAR	100	1092	36,28	380,27	8,68	9479	25,44
MURU	100	1155	38,37	370,63	11,39	13159	26,90
CAR	100	1247	41,43	350,58	9,26	11548	29,05
BMFR	100	1346	44,71	318,28	10,45	14065	31,35

Senaryo 2’de test ortam üzerindeki araç sayıları ve trafik hareketleri Çizelge 4.4’de gösterilmiştir. Senaryo 2 için araç trafiğinin Sumo ortamında oluşturulması ise Şekil 4.20’de gösterilmiştir.

Çizelge 4.4. Birinci model ikinci senaryoda üretilen araç sayısı ve trafik akışı

Başlangıç Noktası	Bitiş Noktası	Başlangıç Süresi	Bitiş Süresi	Araç Sayısı
Kurtuluş Kavşağı	G.M.K bulvarı	0	300	30
Kurtuluş Kavşağı	Altınsoy Cad.	0	300	30
Atatürk Bulvarı	Kurtuluş Kavşağı	0	300	40
Strazburg Cad.	Necatibey Cad.	0	300	40
Atatürk Bulvarı	Fevzi Çakmak Sok.	0	300	30
Atatürk Bulvarı	G.M.K bulvarı	0	300	30

```
<flows>
<flow id="flow1" from="12195038#2" to="68695137#5" begin="0" end="300" no="30" />
<flow id="flow2" from="12195038#0" to="28470416#6" begin="0" end="300" no="30" />
<flow id="flow3" from="68695134#3" to="12195038#0" begin="0" end="300" no="40" />
<flow id="flow4" from="26302861#0" to="26317869#4" begin="0" end="300" no="40" />
<flow id="flow5" from="267158069#0" to="46039706#2" begin="0" end="300" no="30" />
<flow id="flow6" from="267158069#1" to="68695130#4" begin="0" end="300" no="30" />
</flows>
```

Şekil 4.20. Birinci model ikinci senaryoda araç trafiğinin sumo ortamında oluşturulması

İkinci senaryoda, araç sayısı 200 olarak belirlenmiş ve senaryo 2 olarak isimlendirilmiştir. Araçların sayısı ve trafik akışı tüm protokoller için aynı tutulmuştur. Eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.5’de sunulmuştur. PDR oranı yaklaşık % 80 ve throughput miktarı yaklaşık 56 kbps ile en başarılı protokol VNPER olmuştur. VNPER’e en yakın PDR oranına sahip olan protokol yaklaşık % 72 ile BMFR protokolü olmuştur. En düşük E2E gecikme değerlerine sahip olan protokoller de benzer şekilde sırasıyla VNPER ve BMFR olmuştur. Yönlendirme yükü incelendiğinde, en fazla yönlendirme yüküne sahip olan protokol MURU olurken, en az yönlendirme yüküne sahip olan protokol ASTAR olmuştur. VNPER protokolü, ASTAR’dan sonra en az yönlendirme yüküne sahip olan protokoldür. VNPER protokolü, ASTAR protokolünden daha masraflı yönlendirme yüküne sahip olmasına rağmen, en verimli sonuçları elde etmiştir. VNPER protokolü, yaklaşık olarak 48000 adet yönlendirme paketi üretirken, ASTAR protokolünde bu sayı yaklaşık olarak 30000’dir.

MURU protokolü ise yaklaşık 59000 yönlendirme paketi üreterek, yaklaşık % 29 oranıyla en fazla yönlendirme yüküne sahip olan protokol olmuştur.

Çizelge 4.5. Birinci model ikinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Araç Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	200	2422	80,47	156,98	19,78	47906	56,28
ASTAR	200	1920	63,79	190,17	15,54	29844	36,71
MURU	200	2025	67,27	184,62	29,15	59023	47,45
CAR	200	2088	69,37	174,43	20,73	43287	44,66
BMFR	200	2171	72,13	160,32	23,65	51341	55,20

Senaryo 3'de test ortam üzerindeki araç sayıları ve trafik hareketleri Çizelge 4.6'da gösterilmiştir. Senaryo 3 için araç trafiğinin Sumo ortamında oluşturulması ise Şekil 4.21'de gösterilmiştir.

Çizelge 4.6. Birinci model üçüncü senaryoda üretilen araç sayısı ve trafik akışı

Başlangıç Noktası	Bitiş Noktası	Başlangıç Süresi	Bitiş Süresi	Araç Sayısı
Kurtuluş Kavşağı	G.M.K bulvarı	0	300	50
Kurtuluş Kavşağı	Altınsoy Cad.	0	300	50
Atatürk Bulvarı	Kurtuluş Kavşağı	0	300	50
Strazburg Cad.	Necatibey Cad.	0	300	50
Atatürk Bulvarı	Fevzi Çakmak Sok.	0	300	50
Atatürk Bulvarı	G.M.K bulvarı	0	300	50

```
<flows>
<flow id="flow1" from="12195038#2" to="68695137#5" begin="0" end="300" no="50" />
<flow id="flow2" from="12195038#0" to="28470416#6" begin="0" end="300" no="50" />
<flow id="flow3" from="68695134#3" to="12195038#0" begin="0" end="300" no="50" />
<flow id="flow4" from="26302861#0" to="26317869#4" begin="0" end="300" no="50" />
<flow id="flow5" from="267158069#0" to="46039706#2" begin="0" end="300" no="50" />
<flow id="flow6" from="267158069#1" to="68695130#4" begin="0" end="300" no="50" />
</flows>
```

Şekil 4.21. Birinci model üçüncü senaryoda araç trafiğinin sumo ortamında oluşturulması

Üçüncü senaryoda, araç miktarı 300 olarak belirlenmiştir ve senaryo 3 olarak isimlendirilmiştir. Araçların sayısı ve trafik akışı tüm protokoller için aynı tutulmuştur. Eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur.

Elde edilen sonuçlar Çizelge 4.7’de sunulmuştur. Diğer senaryolarda olduğu gibi, en çok PDR ve throughput değerine sahip olan protokol VNPER olmuştur. Benzer şekilde, en düşük E2E gecikme değerine sahip olan protokol de VNPER olmuştur. Geriye kalan diğer protokoller arasında en düşük E2E gecikme değerine ve en yüksek paket teslim etme oranına sahip olan protokoller sırası ile BMFR, CAR, MURU ve ASTAR olmuştur. En fazla yönlendirme yüküne sahip olan protokol ise MURU olmuştur. Diğer protokoller arasında en yüksek yönlendirme yüküne sahip olan protokoller ise sırasıyla BMFR, CAR, VNPER ve ASTAR olmuştur. ASTAR protokolü en düşük yönlendirme yüküne sahip olmasına rağmen, en düşük verime sahip olan protokol olduğu sonucuna ulaşılmıştır. VNPER protokolünün yönlendirme yükü ASTAR protokolüne göre daha fazla olmasına rağmen, diğer metriklere göre daha başarılı sonuçlar elde etmiştir.

Çizelge 4.7. Birinci model üçüncü senaryodan elde edilen deneysel sonuçlar

Protokoller	Araç Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yükü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	300	2209	73,39	186,71	24,43	53971	50,13
ASTAR	300	1528	50,76	225,75	20,12	30743	34,32
MURU	300	1713	56,91	218,59	34,12	58449	39,57
CAR	300	1826	60,66	207,92	26,89	49105	40,65
BMFR	300	1936	64,32	194,72	28,91	55968	48,56

Senaryo 4’de test ortam üzerindeki araç sayıları ve trafik hareketleri Çizelge 4.8’de gösterilmiştir. Senaryo 4 için araç trafiğinin Sumo ortamında oluşturulması ise Şekil 4.22’de gösterilmiştir.

Çizelge 4.8. Birinci model dördüncü senaryoda üretilen araç sayısı ve trafik akışı

Başlangıç Noktası	Bitiş Noktası	Başlangıç Süresi	Bitiş Süresi	Araç Sayısı
Kurtuluş Kavşağı	G.M.K bulvarı	0	300	60
Kurtuluş Kavşağı	Altınsoy Cad.	0	300	60
Atatürk Bulvarı	Kurtuluş Kavşağı	0	300	80
Strazburg Cad.	Necatibey Cad.	0	300	80
Atatürk Bulvarı	Fevzi Çakmak Sok.	0	300	60
Atatürk Bulvarı	G.M.K bulvarı	0	300	60

```

<flows>
<flow id="flow1" from="12195038#2" to="68695137#5" begin="0" end="300" no="60" />
<flow id="flow2" from="12195038#0" to="28470416#6" begin="0" end="300" no="60" />
<flow id="flow3" from="68695134#3" to="12195038#0" begin="0" end="300" no="80" />
<flow id="flow4" from="26302861#0" to="26317869#4" begin="0" end="300" no="80" />
<flow id="flow5" from="267158069#0" to="46039706#2" begin="0" end="300" no="60" />
<flow id="flow6" from="267158069#1" to="68695130#4" begin="0" end="300" no="60" />
</flows>

```

Şekil 4.22. Birinci model dördüncü senaryoda araç trafiğinin sumo ortamında oluşturulması

Dördüncü senaryoda, araç miktarı 400 olarak belirlenmiştir ve senaryo 4 olarak isimlendirilmiştir. Araçların sayısı ve trafik akışı tüm protokoller için aynı tutulmuştur. Eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.9’da sunulmuştur. PDR, throughput ve E2E gecikme değeri en verimli protokol VNPER olmuştur. Geriye kalan diğer protokoller arasında en verimli protokol ise BMFR olmuştur. En fazla yönlendirme yüküne sahip olan protokol MURU olurken en az yönlendirme yüküne sahip olan protokol ise ASTAR olmuştur. VNPER protokolü ise ASTAR protokolünden sonra en az yönlendirme yüküne sahip olan protokol olmuştur.

Çizelge 4.9. Birinci model dördüncü senaryodan elde edilen deneysel sonuçlar

Protokoller	Araç Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yükü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	400	1814	60,26	219,14	30,93	56107	41,17
ASTAR	400	853	28,34	272,64	25,64	21871	22,07
MURU	400	1160	38,54	254,81	42,35	49120	28,10
CAR	400	1321	43,89	242,55	32,65	43135	31,96
BMFR	400	1488	49,44	230,84	35,67	53082	34,94

Senaryo 5’de test ortam üzerindeki araç sayıları ve trafik hareketleri Çizelge 4.10’da gösterilmiştir. Senaryo 5 için araç trafiğinin Sumo ortamında oluşturulması ise Şekil 4.23’de gösterilmiştir.

Çizelge 4.10. Birinci model beşinci senaryoda üretilen araç sayısı ve trafik akışı

Başlangıç Noktası	Bitiş Noktası	Başlangıç Süresi	Bitiş Süresi	Araç Sayısı
Kurtuluş Kavşağı	G.M.K bulvarı	0	300	75
Kurtuluş Kavşağı	Altınsoy Cad.	0	300	75
Atatürk Bulvarı	Kurtuluş Kavşağı	0	300	100
Strazburg Cad.	Necatibey Cad.	0	300	100
Atatürk Bulvarı	Fevzi Çakmak Sok.	0	300	75
Atatürk Bulvarı	G.M.K bulvarı	0	300	75

```

<flows>
<flow id="flow1" from="12195038#2" to="68695137#5" begin="0" end="300" no="75" />
<flow id="flow2" from="12195038#0" to="28470416#6" begin="0" end="300" no="75" />
<flow id="flow3" from="68695134#3" to="12195038#0" begin="0" end="300" no="100" />
<flow id="flow4" from="26302861#0" to="26317869#4" begin="0" end="300" no="100" />
<flow id="flow5" from="267158069#0" to="46039706#2" begin="0" end="300" no="75" />
<flow id="flow6" from="267158069#1" to="68695130#4" begin="0" end="300" no="75" />
</flows>

```

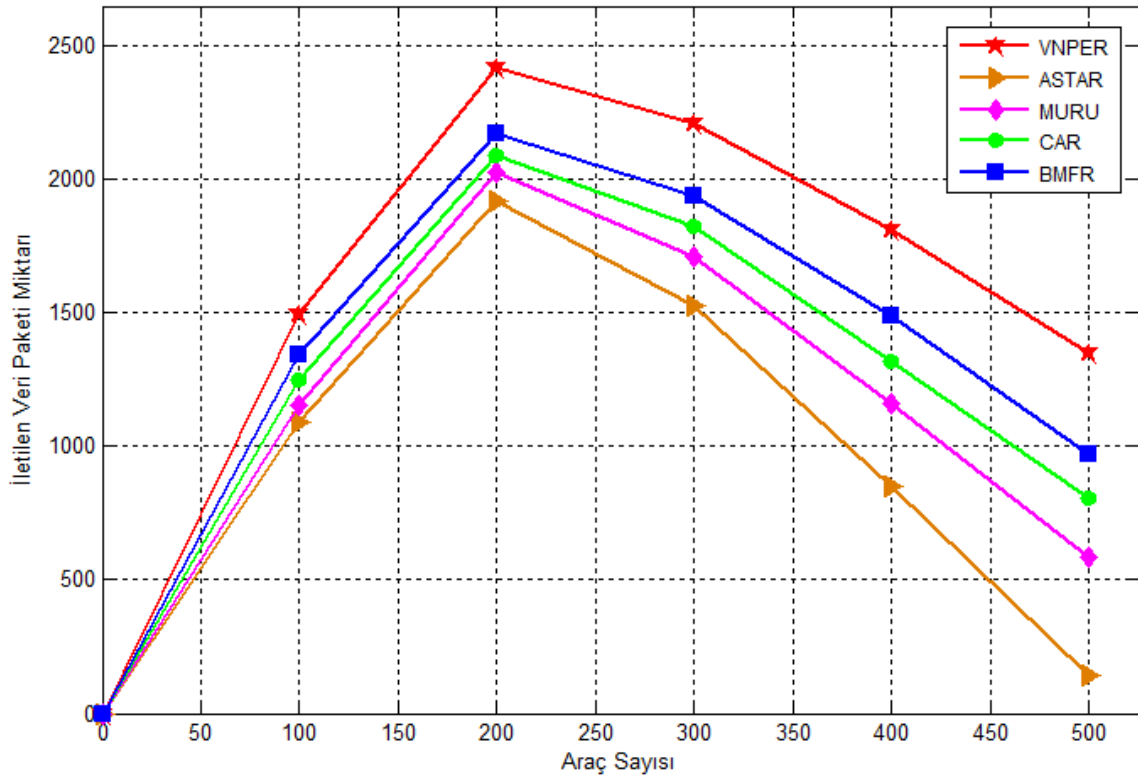
Şekil 4.23. Birinci model beşinci senaryoda araç trafiğinin sumo ortamında oluşturulması

Beşinci senaryoda, araç miktarı 500 olarak belirlenmiştir ve senaryo 5 olarak isimlendirilmiştir. Araçların sayısı ve trafik akışı tüm protokoller için aynı tutulmuştur. Eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.11’de sunulmuştur. PDR, throughput ve E2E gecikme metriklerine göre en başarılı olan protokol VNPER olmuştur. Aynı metrikler dikkate alındığında, ikinci en verimli protokol ise BMFR olmuştur. En fazla yönlendirme yüküne sahip olan protokol ise MURU olurken, en düşük yönlendirme yüküne sahip olan protokol ASTAR olmuştur.

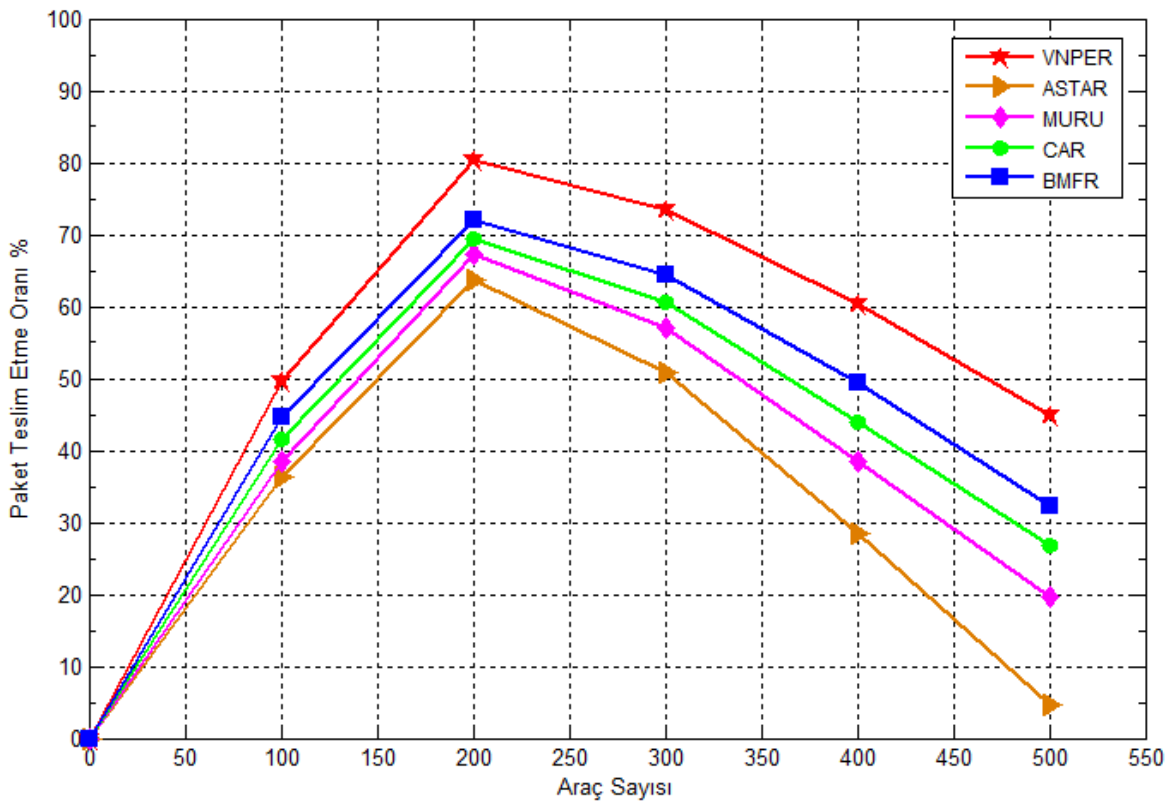
Çizelge 4.11. Birinci model beşinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Araç Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yükü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	500	1351	44,88	262,13	40,14	54228	30,66
ASTAR	500	141	4,68	336,78	33,58	4735	3,59
MURU	500	588	19,53	299,80	53,18	31267	14,85
CAR	500	805	26,74	284,24	42,26	34019	20,34
BMFR	500	970	32,23	268,61	45,85	44475	22,78

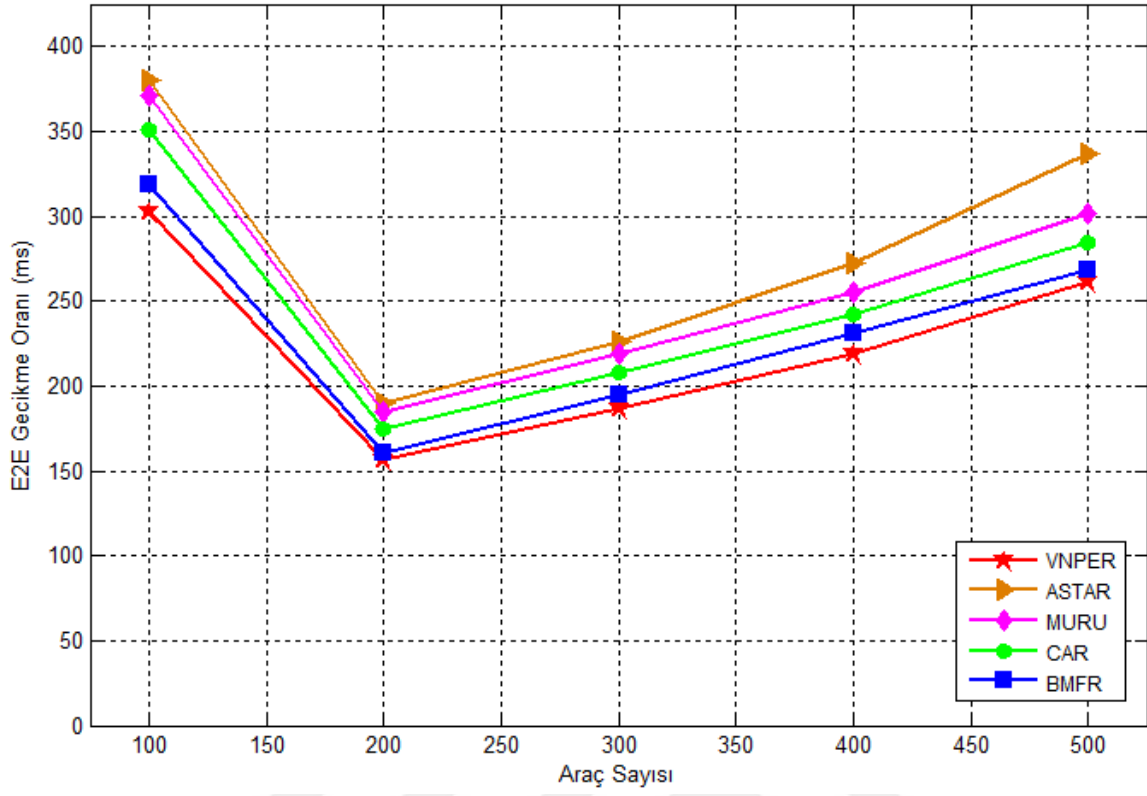
Uygulama sonuçlarından anlaşılacağı gibi, PDR, E2E ve throughput değeri en başarılı protokol VNPER olmuştur. Aynı metriklere göre, en verimsiz protokol ise ASTAR olmuştur. Ancak, en az yönlendirme yüküne sahip olan protokol de ASTAR'dır. En fazla yönlendirme yüküne sahip olan protokol de MURU olmuştur. Özetle, ASTAR protokolüne göre, VNPER protokolü masraflı olmasına rağmen en verimli protokoldür. Diğer protokoller ile kıyaslandığında ise, VNPER protokolü daha az yönlendirme yüküne sahiptir. Ayrıca, üretilen trafik miktarına göre, yönlendirme yükü en az değişkenlik gösteren protokol VNPER'dir. Üretilen trafik miktarı arttıkça, yönlendirme yükünde de artış olmaktadır. Ancak, en az artış oranı VNPER protokolündedir. Bundan dolayı, trafik yükü arttıkça VNPER protokolünün yönlendirme yükü diğer protokollere göre daha az olmaktadır. Şekil 4.24, 4.25, 4.26, 4.27, 4.28 ve 4.29'da birinci modelden elde edilen deneysel sonuçların farklı metriklere göre kıyaslanması sonucu ortaya çıkan grafiksel sonuçlar gösterilmiştir. Şekil 4.24'de kaynaktan hedefe başarılı şekilde ulaştırılan veri paketi miktarının kıyaslanması gösterilmiştir. Şekil 4.25'de protokollerin kaynaktan hedefe doğru paket teslim etme oranlarının kıyaslanması gösterilmiştir. Şekil 4.26'da bir veri paketinin kaynaktan hedefine doğru hangi oranda ortalama gecikme değeri ile ulaştığı karşılaştırmalı olarak gösterilmiştir. Şekil 4.27'de bir veri paketinin iletilmesi için üretilmesi gereken yönlendirme paketi oranı kıyaslamalı olarak gösterilmiştir. Şekil 4.28'de benzetim süresi boyunca üretilen toplam yönlendirme paketi sayıları karşılaştırmalı olarak gösterilmiştir. Şekil 4.29'da benzetim süresi boyunca protokollerin toplam işlem yapabilme kapasitesi karşılaştırmalı olarak gösterilmiştir.



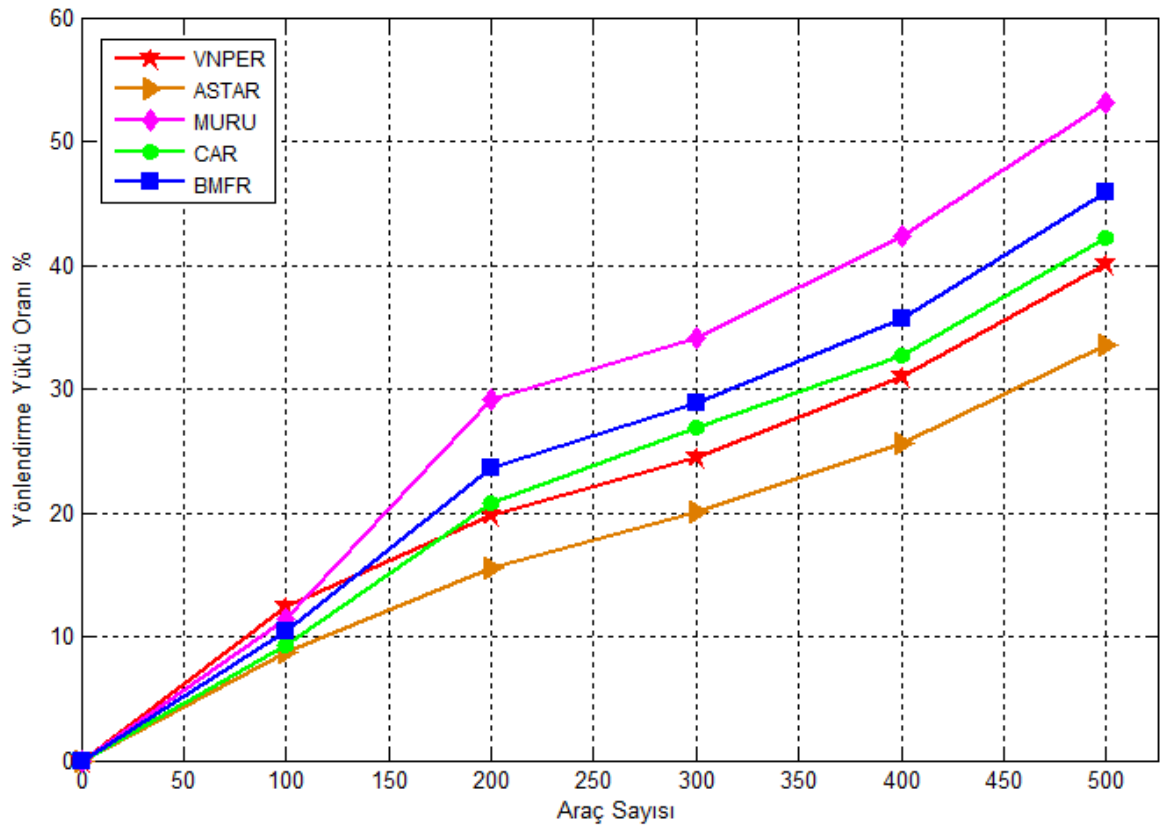
Şekil 4.24. Birinci model için iletilen veri paketi miktarının kıyaslanması



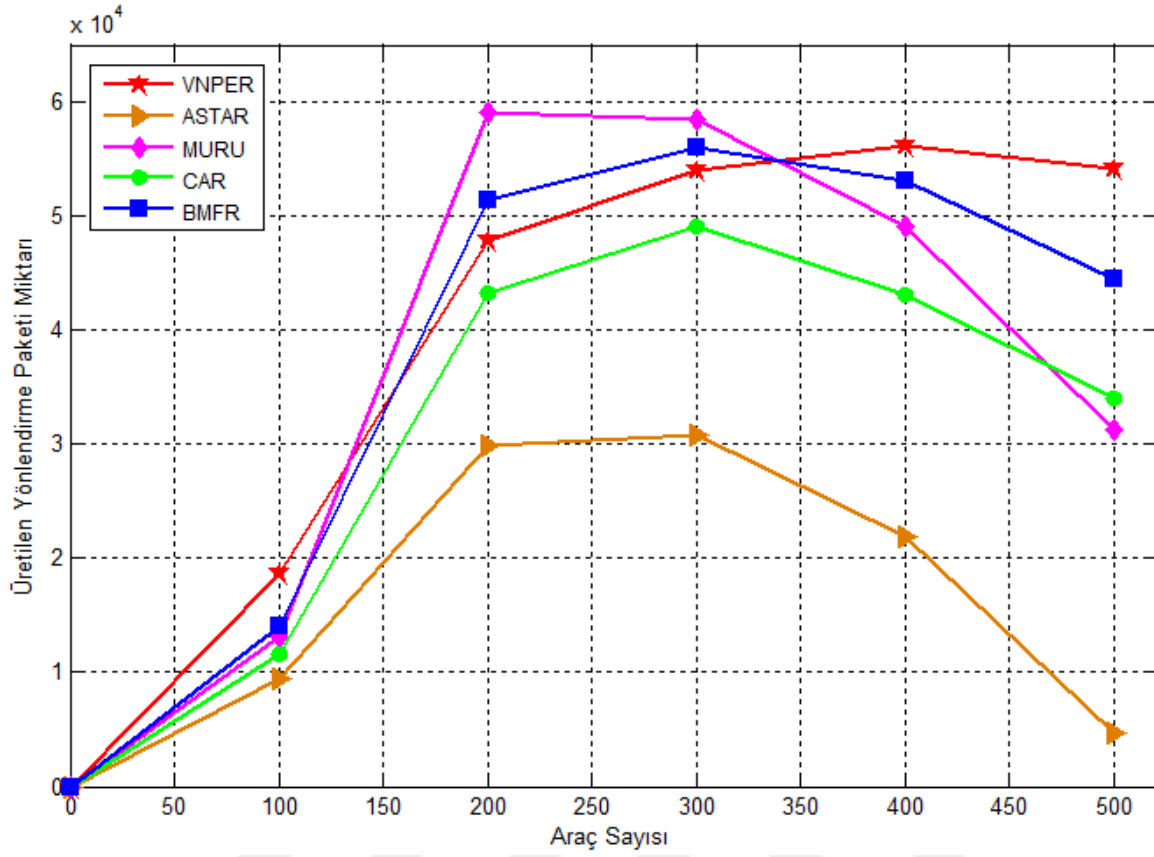
Şekil 4.25. Birinci model için paket teslim etme oranının kıyaslanması



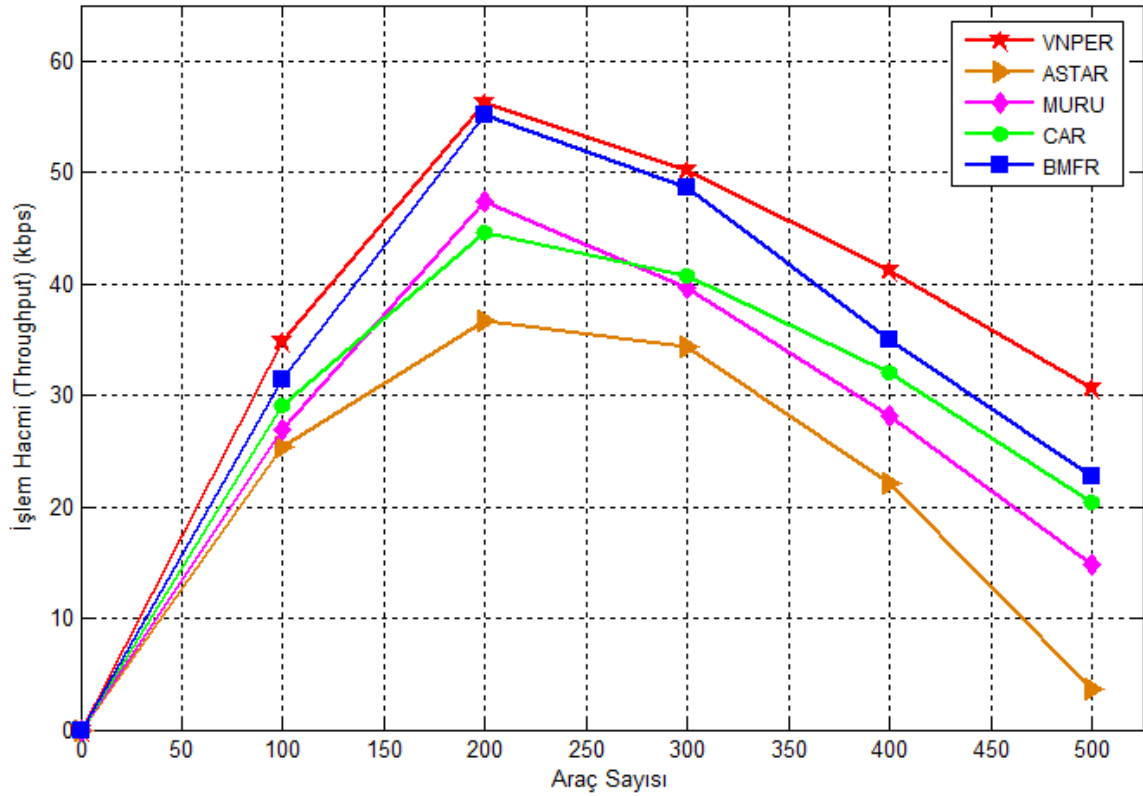
Şekil 4.26. Birinci model için uçtan uca gecikme oranının kıyaslanması



Şekil 4.27. Birinci model için yönlendirme yükü oranının kıyaslanması



Şekil 4.28. Birinci model için yönlendirme yükü paketi miktarının kıyaslanması



Şekil 4.29. Birinci model için işlem hacmi miktarının kıyaslanması

4.4.2. İkinci modelden elde edilen deneysel sonuçlar

İkinci benzetim modelinde, 260 adet araç farklı noktalardan başlayarak farklı yönlere doğru hareket edecek şekilde 300 saniye boyunca araç trafiği oluşturulmuştur. Araçlar arasında ise veri trafiği oluşturularak kaynak araçtan hedef araca doğru veri iletilmesi sağlanmıştır. Araç sayısı sabit tutulup veri trafik miktarı artırılmıştır. Test ortamında kullanılan NS2 parametreleri Çizelge 4.12’de gösterilmiştir. Oluşturulan veri trafiğinin hedef düğüme ne oranda ulaştığı, ne kadar sürede ulaştığı gibi ölçümler yapılarak protokollerin verimliliği test edilmiştir.

Çizelge 4.12. İkinci modelde kullanılan parametreler

Mobil düğüm sayısı	260 düğüm
Kanal Tipi	WirelessChannel
Yayılm Modeli	TwoRayGround
Interface Tipi	WirelessPhy
MAC Tipi	802_11
Kuyruk Tipi	DropTail/PriQueue
Anten tipi	OmniAntenna
Bağlantı Katmanı Tipi	LL
Kuyruk uzunluğu	50 paket
X Koordinat uzunluğu	16559 birim
Y Koordinat uzunluğu	6656 birim
Benzetim süresi	300 sn

Uygulama haritası üzerinde bulunan 6 tane farklı noktadan farklı hedeflere doğru araç trafiği başlatılmıştır. Örneğin, Kurtuluş kavşağından Gazi Mustafa Kemal Bulvarına doğru 0. saniyeden 300. saniyeye kadar 40 adet araç trafiği oluşturulmuştur. Uygulama ortamında oluşturulan araç trafik hareketleri Çizelge 4.13’de gösterilmiştir. Araç trafiğinin Sumo ortamında oluşturulması ise Şekil 4.30’da gösterilmiştir.

Çizelge 4.13. İkinci modelde üretilen araç sayısı ve trafik akışı

Başlangıç Noktası	Bitiş Noktası	Başlangıç Süresi	Bitiş Süresi	Araç Sayısı
Kurtuluş Kavşağı	G.M.K bulvarı	0	300	40
Kurtuluş Kavşağı	Altınsoy Cad.	0	300	40
Atatürk Bulvarı	Kurtuluş Kavşağı	0	300	50
Strazburg Cad.	Necatibey Cad.	0	300	50
Atatürk Bulvarı	Fevzi Çakmak Sok.	0	300	40
Atatürk Bulvarı	G.M.K bulvarı	0	300	40

```

</flows>
<flow id="flow1" from="12195038#2" to="68695137#5" begin="0" end="300" no="40" />
<flow id="flow2" from="12195038#0" to="28470416#6" begin="0" end="300" no="40" />
<flow id="flow3" from="68695134#3" to="12195038#0" begin="0" end="300" no="50" />
<flow id="flow4" from="26302861#0" to="26317869#4" begin="0" end="300" no="50" />
<flow id="flow5" from="267158069#0" to="46039706#2" begin="0" end="300" no="40" />
<flow id="flow6" from="267158069#1" to="68695130#4" begin="0" end="300" no="40" />
</flows>

```

Şekil 4.30. İkinci model birinci senaryoda araç trafiğinin sumo ortamında oluşturulması

İlk senaryoda, araçlar arasında veri transferinde kullanılmak üzere 839 adet paketin olacağı bir veri trafiği üretilmiştir ve senaryo 1 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.14'de sunulmuştur. Protokoller, aynı senaryo üzerinde çalıştırıldığı için aynı sayıda veri paketi üretilmiştir. Üretilen 839 adet veri paketinden 804 adeti VNPER protokolü ile hedefine başarılı şekilde ulaştırılmıştır. Paket teslim etme oranı ve throughput değeri en başarılı protokol VNPER olmuştur. E2E gecikme değeri en başarılı olan protokoller ise MURU ve VNPER olmuştur. Ancak, PDR ve throughput değeri VNPER protokolünün gerisinde kalmıştır. Yönlendirme yükü en az olan protokol CAR ve BMFR olurken, en fazla yönlendirme yüküne sahip olan protokol VNPER olmuştur. VNPER protokolü bu açıdan bakıldığında diğer protokollere kıyasla yüksek oranda PDR ile başarılı şekilde veri iletimini gerçekleştirmiştir. Veri iletimini gerçekleştirmek için diğer protokollere oranla daha fazla yönlendirme paketi üretmiştir.

Çizelge 4.14. İkinci model birinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Üretilen Paket (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	839	804	95,83	45,95	13,12	10554	73,97
ASTAR	839	703	83,79	48,26	5,76	4051	64,55
MURU	839	756	90,11	45,71	4,71	3565	69,54
CAR	839	783	93,32	51,09	4,14	3243	72,06
BMFR	839	795	94,75	50,44	4,23	3351	68,85

İkinci senaryoda, araçlar arasında veri transferinde kullanılmak üzere 1562 adet paketin olacağı bir veri trafiği üretilmiştir ve senaryo 2 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.15’de sunulmuştur. PDR oranı yaklaşık % 87 ve throughput miktarı yaklaşık 61 Kbps ile en başarılı protokol VNPER olmuştur. VNPER’e en yakın PDR oranına sahip olan protokol yaklaşık % 85 ile BMFR protokolü olmuştur. En düşük E2E gecikme değerine sahip protokol ise yaklaşık 71 ms ile VNPER olmuştur. BMFR protokolünün E2E gecikme değeri ise yaklaşık 74 ms’dir. Yönlendirme yükü incelendiğinde, en fazla yönlendirme yüküne sahip olan protokol VNPER olurken, en az yönlendirme yüküne sahip olan protokol ASTAR olmuştur. VNPER protokolü masraflı bir yönlendirme yüküne sahip olmasına rağmen, en verimli sonuçları elde etmiştir. VNPER protokolü, yönlendirme paketlerini en çok ağda dolaştıran protokoldür. 20853 tane yönlendirme paketi ağdaki düğümler arasında dolaştırılmıştır.

Çizelge 4.15. İkinci model ikinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Üretilen Paket (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	1562	1369	87,65	71,29	15,23	20853	61,44
ASTAR	1562	1164	74,52	80,47	9,16	10665	52,24
MURU	1562	1255	80,34	75,58	11,16	14009	56,32
CAR	1562	1301	83,29	78,41	10,67	13886	58,39
BMFR	1562	1329	85,08	74,22	9,36	12447	55,40

Üçüncü senaryoda, araçlar arasında veri transferinde kullanılmak üzere 2667 adet paketin olacağı bir veri trafiği üretilmiştir ve senaryo 3 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.16’da sunulmuştur. Diğer senaryolarda olduğu gibi, en çok PDR ve throughput değerine sahip olan protokol VNPER olmuştur. En düşük E2E gecikme değerine sahip olan protokol de benzer şekilde VNPER olmuştur. En düşük PDR oranına sahip olan protokol ise ASTAR olmuştur. VNPER dışında geriye kalan diğer protokoller arasında en düşük E2E gecikme değerine sahip olan protokoller sırası ile BMFR, CAR, MURU ve ASTAR olmuştur. En fazla yönlendirme yüküne sahip olan protokoller ise sırasıyla MURU ve VNPER olmuştur. En düşük yönlendirme yüküne sahip olan protokol ise ASTAR olmuştur. Bu açıdan bakıldığında en az yönlendirme masrafına sahip olan protokol ASTAR’dır. VNPER diğer protokollere göre masraflı bir protokol olması rağmen, en başarılı protokol olmuştur.

Çizelge 4.16. İkinci model üçüncü senaryodan elde edilen deneysel sonuçlar

Protokoller	Üretilen Paket (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yükü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	2667	2090	78,36	120,38	19,34	40431	54,94
ASTAR	2667	1606	60,22	147,79	16,27	26131	42,21
MURU	2667	1794	67,27	136,23	19,67	35278	47,16
CAR	2667	1873	70,23	132,57	18,77	35171	49,23
BMFR	2667	1961	73,53	128,63	17,88	35077	51,55

Dördüncü senaryoda, araçlar arasında veri transferinde kullanılmak üzere 4115 adet paketin olacağı bir veri trafik üretilmiştir ve senaryo 4 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.17’de sunulmuştur. PDR ve throughput değeri en verimli protokol VNPER olmuştur. E2E gecikme değeri en düşük protokol VNPER olurken, en fazla E2E değerine sahip olan protokol ise ASTAR olmuştur. Diğer protokollerde en verimli E2E değerine sahip olan protokoller sırasıyla BMFR, CAR ve MURU olmuştur. En fazla yönlendirme yüküne sahip olan protokol MURU olurken en az yönlendirme yüküne sahip olan protokol ise ASTAR olmuştur. VNPER protokolünün

yönlendirme yükündeki artış oranı diğer protokollere oranla daha az olmuştur. ASTAR protokolünden sonra en az yönlendirme yüküne sahip olan protokol VNPER olmuştur.

Çizelge 4.17. İkinci model dördüncü senaryodan elde edilen deneysel sonuçlar

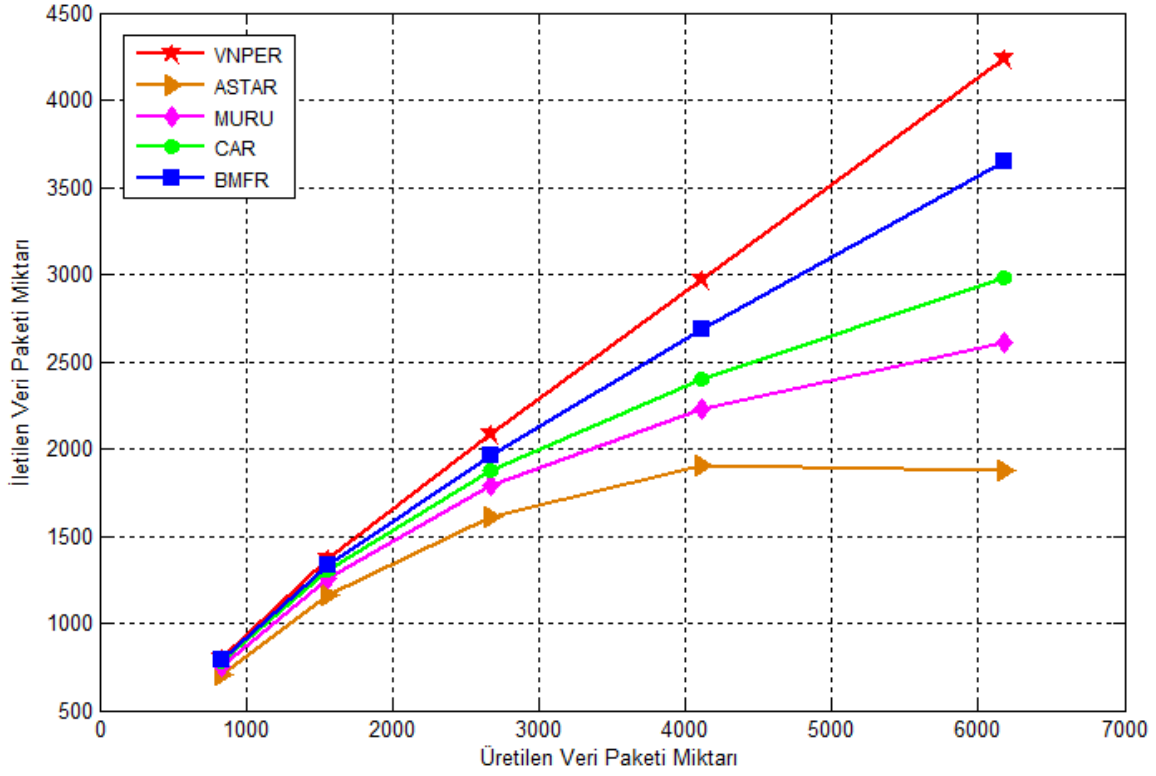
Protokoller	Üretilen Paket (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yükü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	4115	2969	72,15	235,50	27,51	81680	50,58
ASTAR	4115	1903	46,24	308,43	23,48	44693	32,42
MURU	4115	2229	54,17	291,53	36,96	82387	37,98
CAR	4115	2401	58,35	273,87	29,58	71027	40,15
BMFR	4115	2686	65,27	265,46	31,15	83676	45,76

Beşinci senaryoda, araçlar arasında veri transferinde kullanılmak üzere 6172 adet paketin olacağı bir veri trafiği üretilmiştir ve senaryo 5 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği üretilmiş ve veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.18’de sunulmuştur. PDR, throughput ve E2E gecikme değerleri en başarılı olan protokol VNPER olmuştur. VNPER’den sonra en düşük E2E değerine sahip protokol BMFR olurken, en yüksek gecikme değerine sahip olan protokol ASTAR olmuştur. PDR oranı en başarılı ikinci protokol de BMFR olmuştur. En düşük yönlendirme yüküne sahip olan protokol ASTAR olmuştur. ASTAR’dan sonra en düşük yönlendirme yüküne sahip olan protokol ise VNPER olmuştur. En fazla yönlendirme yüküne sahip olan protokol ise MURU olmuştur. CAR ve BMFR protokollerinin yönlendirme yükü oranları ise birbirlerine yakın değerlerdedir.

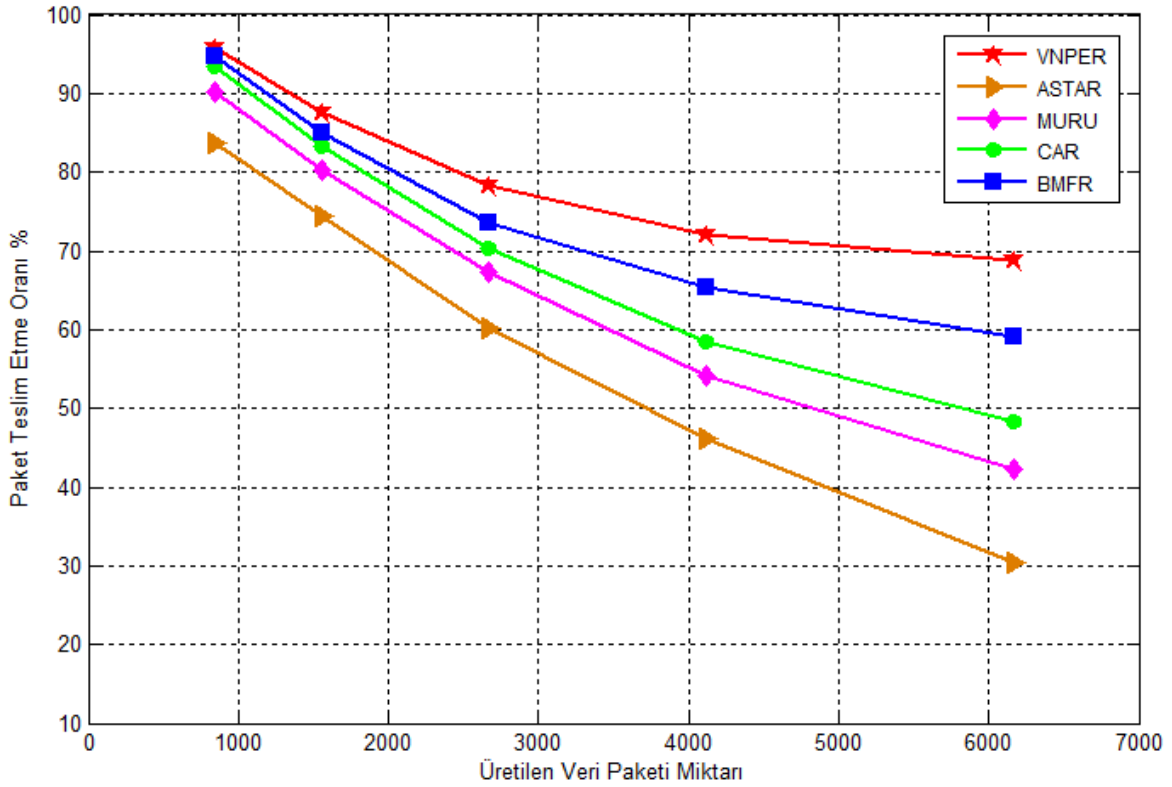
Çizelge 4.18. İkinci model beşinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Üretilen Paket (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yükü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	6172	4241	68,71	471,27	35,78	151763	48,17
ASTAR	6172	1879	30,44	623,4	34,69	65185	21,34
MURU	6172	2609	42,27	592,73	57,19	149219	29,09
CAR	6172	2978	48,25	568,45	45,38	135136	33,82
BMFR	6172	3652	59,17	535,76	47,32	172814	41,48

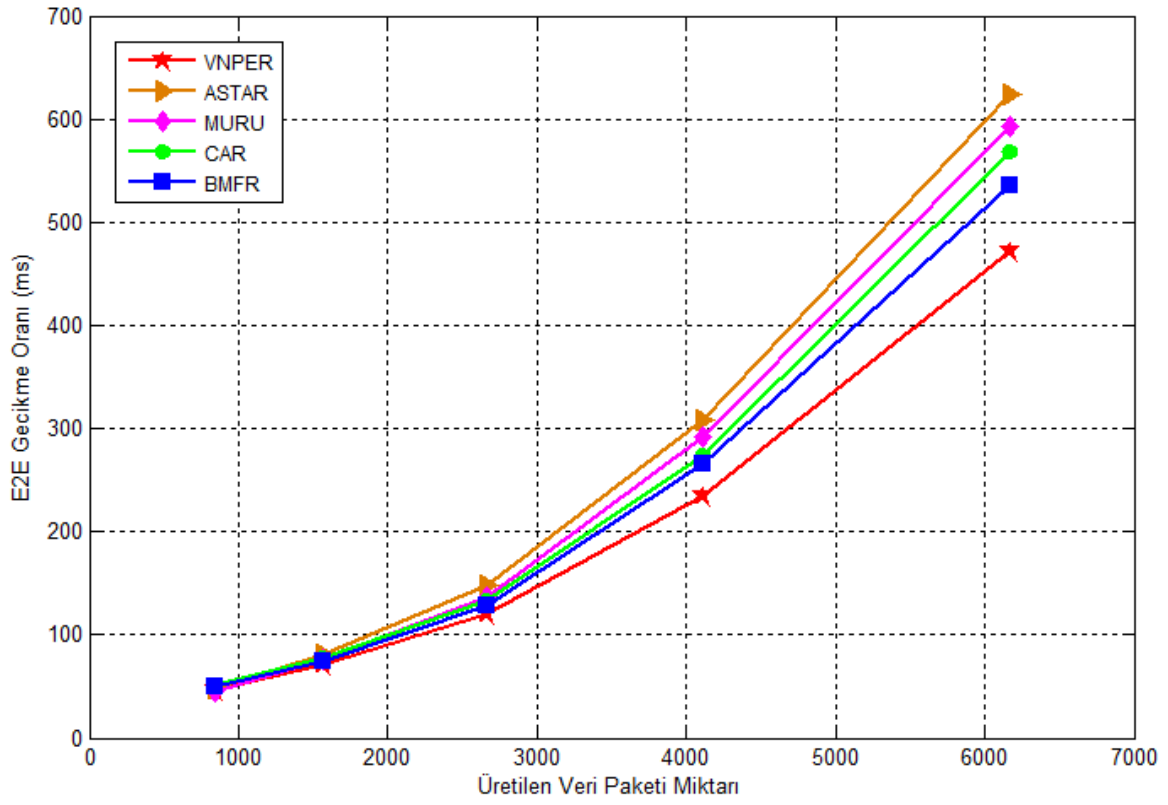
Deneysel sonuçlardan anlaşılacağı gibi, PDR, E2E gecikme ve throughput değeri en başarılı protokol VNPER olmuştur. Aynı metrikler göz önünde bulundurulduğunda VNPER protokolünden sonra en başarılı protokol BMFR olmuştur. En fazla yönlendirme yüküne sahip olan protokol ilk iki senaryo için VNPER olmasına rağmen, PDR, throughput ve E2E gecikme değeri en verimli olan protokoldür. Ayrıca, son üç senaryoda yönlendirme yükü en az olan protokol ASTAR'dan sonra VNPER olmuştur. Ağdaki trafik miktarı arttıkça VNPER protokolünün yönlendirme yükü oranı diğer protokollere göre daha verimli bir hale gelmiştir. Trafik miktarındaki artıştan en az etkilenen protokolün VNPER olduğu sonucuna ulaşılmıştır. Rota keşif işlemlerini ağıın büyüklüğünü göz önünde bulundurmadan gerçekleştirmesinden kaynaklanmaktadır. Az trafiğin olduğu ortamlarda ise kabul edilebilir oranlarda yönlendirme yüküne sahiptir. Ancak ağdaki trafiği arttıkça çok daha başarılı sonuçlar elde etmektedir. Özetle, diğer protokollere göre, ilk iki senaryoda VNPER protokolü masraflı olmasına rağmen en verimli protokoldür. Ayrıca, üretilen trafik miktarına göre, yönlendirme yükü en az değişkenlik gösteren protokol VNPER'dir. Üretilen trafik miktarı arttıkça, yönlendirme yükünde de artış olmaktadır. Ancak, en az artış oranı VNPER protokolündedir. Bundan dolayı, trafik yükü arttıkça VNPER protokolünün yönlendirme yükü diğer protokollere göre daha az olmaktadır. Şekil 4.31, 4.32, 4.33, 4.34, 4.35 ve 4.36'da ikinci modelden elde edilen deneysel sonuçların farklı metriklere göre kıyaslanması sonucu ortaya çıkan grafiksel sonuçlar gösterilmiştir. Şekil 4.31'de kaynaktan hedefe başarılı şekilde ulaştırılan veri paketi miktarının kıyaslanması gösterilmiştir. Şekil 4.32'de protokollerin kaynaktan hedefe doğru paket teslim etme oranlarının kıyaslanması gösterilmiştir. Şekil 4.33'de bir veri paketinin kaynaktan hedefine doğru hangi oranda ortalama gecikme değeri ile ulaştığı karşılaştırmalı olarak gösterilmiştir. Şekil 4.34'de bir veri paketinin iletilmesi için üretilmesi gereken yönlendirme paketi oranı kıyaslamalı olarak gösterilmiştir. Şekil 4.35'de benzetim süresi boyunca üretilen toplam yönlendirme paketi sayıları karşılaştırmalı olarak gösterilmiştir. Şekil 4.36'da benzetim süresi boyunca protokollerin toplam işlem yapabilme kapasitesi karşılaştırmalı olarak gösterilmiştir.



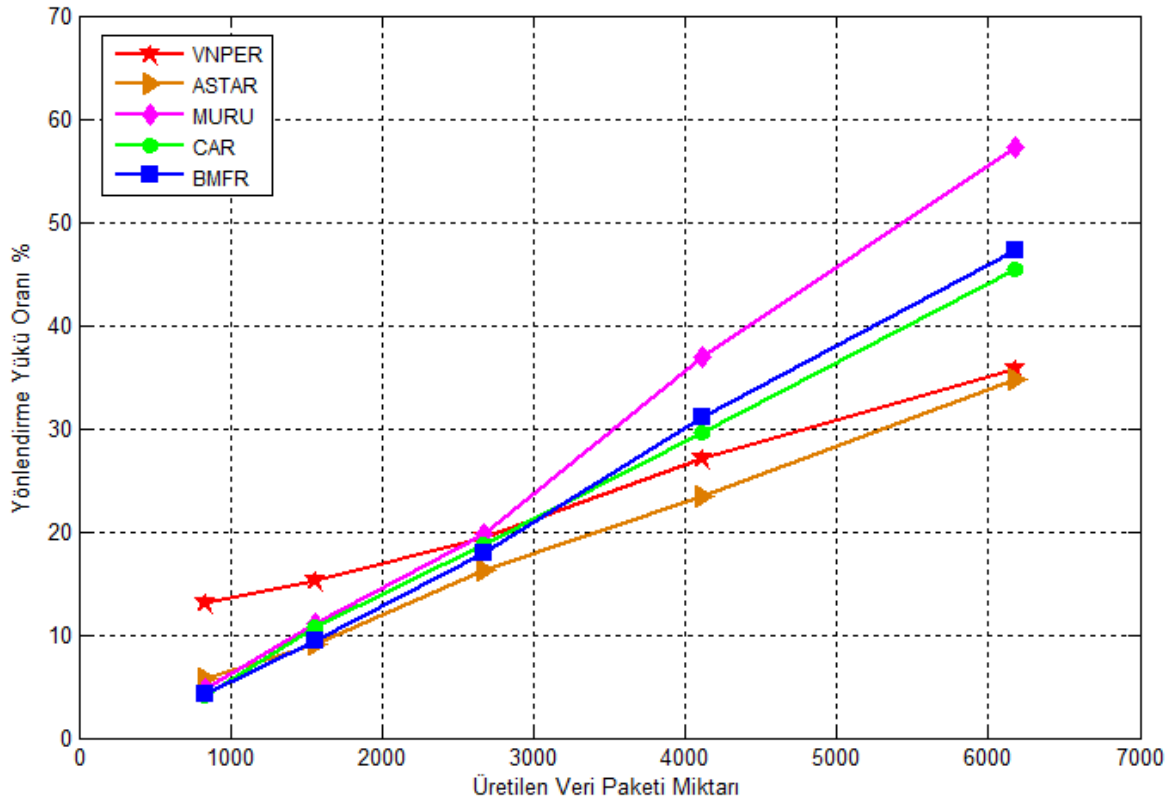
Şekil 4.31. İkinci model için iletilen veri paketi miktarının kıyaslanması



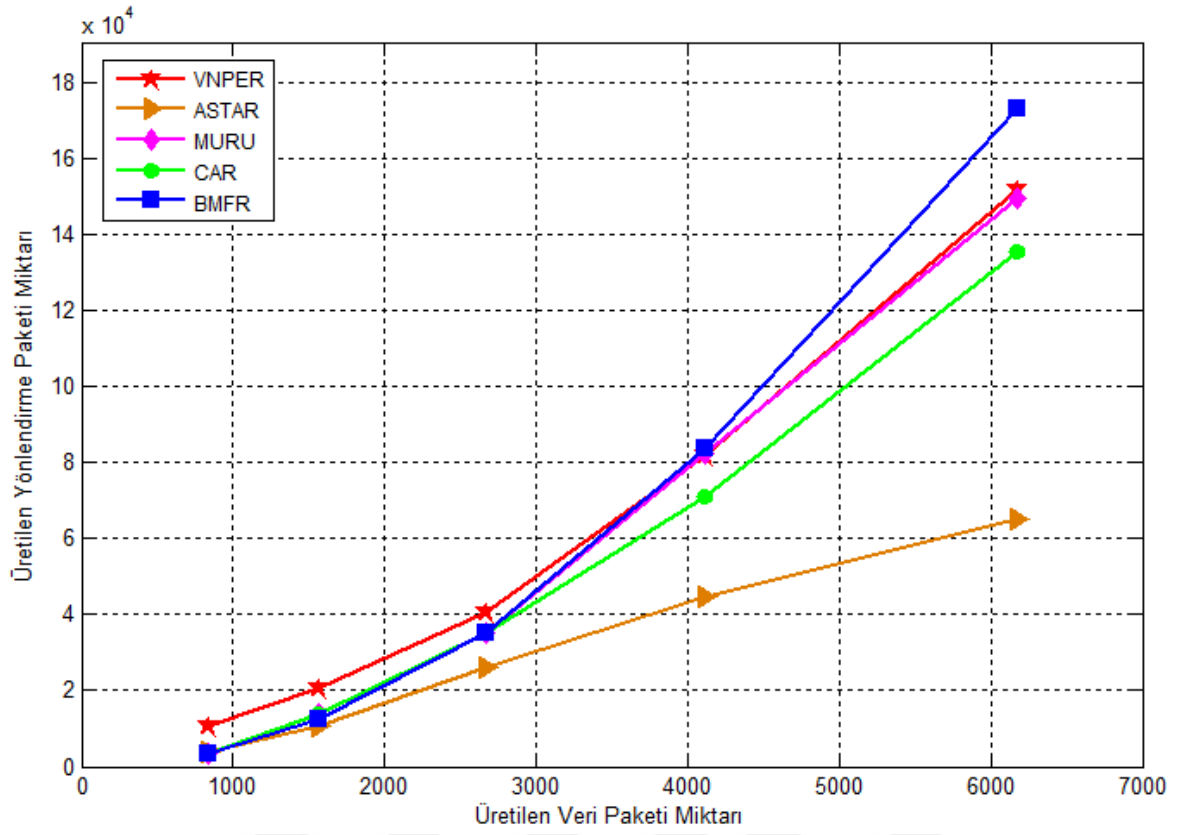
Şekil 4.32. İkinci model için paket teslim etme oranının kıyaslanması



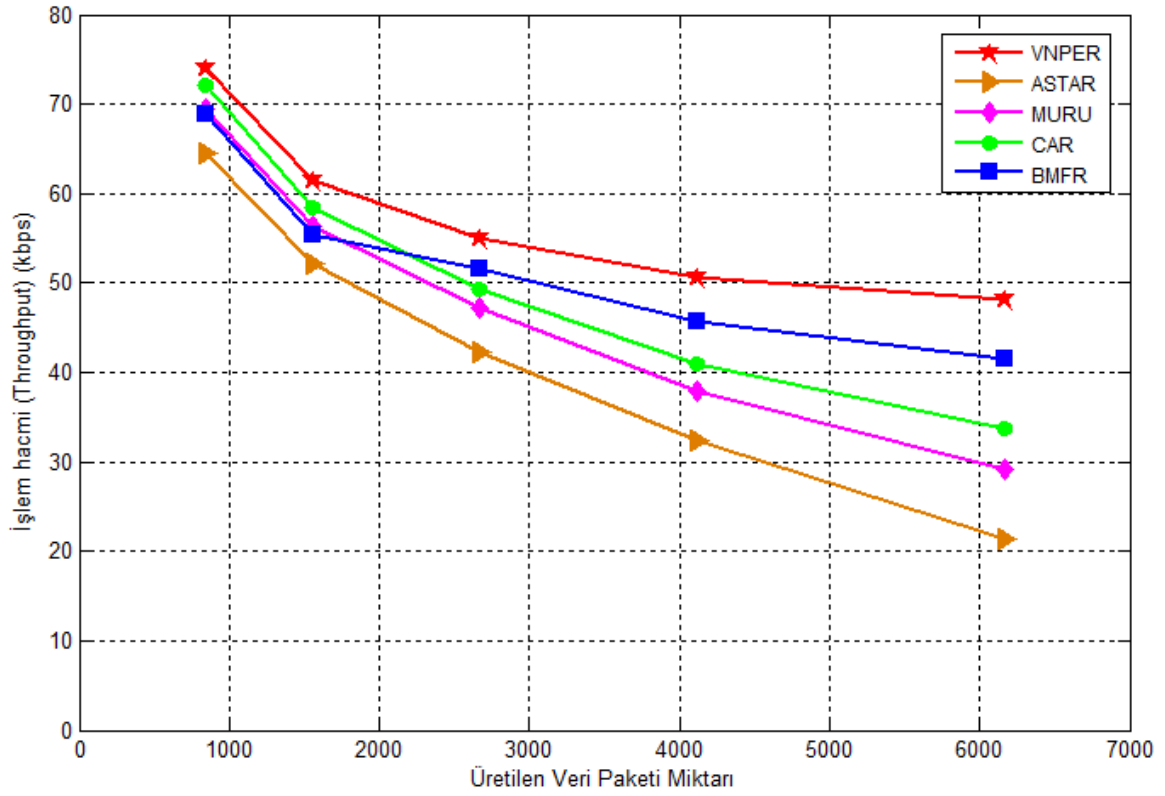
Şekil 4.33. İkinci model için uçtan uca gecikme oranının kıyaslanması



Şekil 4.34. İkinci model için yönlendirme yükü oranının kıyaslanması



Şekil 4.35. İkinci model için üretilen yönlendirme paketi miktarının kıyaslanması



Şekil 4.36. İkinci model için işlem hacmi miktarının kıyaslanması

4.4.3. Üçüncü modelden elde edilen deneysel sonuçlar

Üçüncü benzetim modelinde, benzetim süresi boyunca hem üretilen veri trafik miktarı hem de araç sayısı artırılmıştır. Farklı sayıda araç farklı noktalardan başlayarak farklı yönlerde doğru hareket edecek şekilde 200 saniye boyunca araç trafiği oluşturulmuştur. Araçlar arasında ise veri trafiğinin oluşturulup kaynak araçtan hedef araca doğru iletilmesi sağlanmıştır. Test ortamında kullanılan NS2 parametreleri Çizelge 4.19’da gösterilmiştir. Oluşturulan veri trafiğinin hedef düğüme ne oranda ulaştığı, ne kadar sürede ulaştığı gibi ölçümler yapılarak protokollerin verimliliği test edilmiştir.

Çizelge 4.19. Üçüncü modelde kullanılan parametreler

Kanal Tipi	WirelessChannel
Yayılm Modeli	TwoRayGround
Interface Tipi	WirelessPhy
MAC Tipi	802_11
Kuyruk Tipi	DropTail/PriQueue
Anten tipi	OmniAntenna
Bağlantı Katmanı Tipi	LL
Kuyruk uzunluğu	50 paket
X Koordinat uzunluğu	16559 birim
Y Koordinat uzunluğu	6656 birim
Benzetim süresi	200 sn

Test haritası üzerinde bulunan 6 tane farklı noktadan farklı hedeflere doğru araç trafiği başlatılmıştır. Örneğin, Kurtuluş kavşağından Gazi Mustafa Kemal Bulvarına doğru 0. saniyeden 200. saniyeye kadar sürekli olarak araç trafiği oluşturulmuştur. Ayrıca, araçlardan sürekli olarak veri paketi üretilerek belirli hedeflere doğru veri paketlerinin gönderilmesi sağlanmıştır. Bu iki durumun sonucunda benzetim süresi boyunca sürekli olarak araç miktarında ve veri paketi miktarında zamana göre doğru orantılı olarak artış sağlanmıştır. Test ortamında oluşturulan araç trafik hareketleri Çizelge 4.20’de gösterilmiştir.

Çizelge 4.20. Üçüncü modelde üretilen araç sayısı ve trafik akışı

Başlangıç Noktası	Bitiş Noktası	Başlangıç Süresi	Bitiş Süresi
Kurtuluş Kavşağı	G.M.K bulvarı	0	200
Kurtuluş Kavşağı	Altınsoy Cad.	0	200
Atatürk Bulvarı	Kurtuluş Kavşağı	0	200
Strazburg Cad.	Necatibey Cad.	0	200
Atatürk Bulvarı	Fevzi Çakmak Sok.	0	200
Atatürk Bulvarı	G.M.K bulvarı	0	200

İlk senaryoda, 40 saniyeye kadar hem araç hem de veri trafiği üretilmiştir ve senaryo 1 olarak isimlendirilmiştir. Tüm protokoller için aynı benzetim süresi aralığında aynı miktarda veri ve araç trafiği üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.21’de sunulmuştur. Protokoller, aynı senaryo üzerinde çalıştırıldığı için aynı sayıda veri paketi ve araç sayısı üretilmiştir. 40 saniye boyunca yaklaşık 1039 adet veri paketi ve 48 adet araç üretilmiştir. Üretilen 1039 adet veri paketinden 439 adeti VNPER protokolü ile hedefine başarılı şekilde ulaştırılmıştır. Uçtan uca gecikme değeri ise yaklaşık olarak 188 ms’dir. Paket teslim etme oranı uçtan uca gecikme değeri en başarılı protokol VNPER olmuştur. BMFR protokolü, aynı metrikler dikkate alındığında en başarılı ikinci protokol olmuştur. 407 adet veri paketini hedefine başarılı şekilde ulaştırmıştır ve uçtan uca gecikme değeri yaklaşık olarak 208 ms’dir. Yönlendirme yükü en az olan protokol ASTAR olurken, en fazla yönlendirme yüküne sahip olan protokol VNPER olmuştur. VNPER protokolü başarılı şekilde veri iletimini gerçekleştirmek için diğer protokollere oranla daha fazla yönlendirme paketi üretmiştir. Ancak, ASTAR protokolünün PDR ve E2E metrik değerleri VNPER protokolünün gerisinde kalmıştır. VNPER protokolü bu açıdan bakıldığında kabul edilebilir oranda yönlendirme yükü oranı ile başarılı şekilde veri iletimini gerçekleştirmiştir.

Çizelge 4.21. Üçüncü model birinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Benzetim Süresi (sn)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	0 – 40	439	42,25	188,65	16,14	7086	29,66
ASTAR	0 – 40	347	33,40	252,15	11,62	4032	23,45
MURU	0 – 40	377	36,28	236,44	16,78	6326	25,47
CAR	0 – 40	389	37,44	225,78	14,53	5652	28,78
BMFR	0 – 40	407	39,17	207,91	12,86	5234	27,94

İkinci senaryoda, 80 saniyeye kadar hem araç hem de veri trafiği üretilmiştir ve senaryo 2 olarak isimlendirilmiştir. Tüm protokoller için aynı benzetim süresi aralığında aynı miktarda veri ve araç trafiği üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.22’de sunulmuştur. Protokoller, aynı senaryo üzerinde çalıştırıldığı için aynı sayıda veri paketi ve araç sayısı üretilmiştir. 80 saniye boyunca yaklaşık 1772 adet veri paketi ve 147 adet araç üretilmiştir. Üretilen 1772 adet veri paketinden 1286 adeti VNPER protokolü ile hedefine başarılı şekilde ulaştırılmıştır. Uçtan uca gecikme değeri ise yaklaşık olarak 156 ms’dir. PDR oranı yaklaşık % 72 ile en başarılı protokol VNPER olmuştur. VNPER’e en yakın PDR oranına sahip olan protokol yaklaşık %67 ile BMFR protokolü olmuştur. BMFR protokolünün uçtan uca gecikme değeri ise yaklaşık 165 ms ile VNPER’den sonra en az gecikme değerine sahip protokol olmuştur. Yönlendirme yükü incelendiğinde, en fazla yönlendirme yüküne sahip olan protokol MURU olurken, en az yönlendirme yüküne sahip olan protokol ASTAR olmuştur. VNPER protokolü MURU’dan sonra en fazla yönlendirme yüküne sahip olan protokoldür. VNPER protokolü yaklaşık olarak %20 oranında yönlendirme yüküne sahip iken, ASTAR protokolü yaklaşık olarak %16 oranında yönlendirme yüküne sahiptir. BMFR protokolü ise, ASTAR’dan sonra yaklaşık olarak % 17 oranı ile en az yönlendirme yüküne sahip ikinci protokol olmuştur. VNPER protokolü, MURU protokolü hariç diğer protokollere kıyasla masraflı bir yönlendirme yüküne sahip olmasına rağmen, en verimli sonuçları elde etmiştir.

Çizelge 4.22. Üçüncü model ikinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Benzetim Süresi (sn)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	41 – 80	1286	72,57	155,89	19,82	25488	52,01
ASTAR	41 – 80	1028	58,01	192,15	16,65	17116	40,58
MURU	41 – 80	1067	60,21	180,23	21,78	23239	41,90
CAR	41 – 80	1130	63,77	173,67	19,25	21752	44,70
BMFR	41 – 80	1195	67,44	165,78	17,36	20745	47,27

Üçüncü senaryoda, 120 saniyeye kadar hem araç hem de veri trafiğı üretilmiştir ve senaryo 3 olarak isimlendirilmiştir. Tüm protokoller için aynı benzetim süresi aralığında aynı miktarda veri ve araç trafiğı üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.23’de sunulmuştur. Protokoller, aynı senaryo üzerinde çalıştırıldığı için aynı sayıda veri paketi ve araç sayısı üretilmiştir. 120 saniye boyunca yaklaşık 2554 adet veri paketi ve 243 adet araç üretilmiştir. Üretilen 2524 adet veri paketinden 2237 adeti VNPER protokolü ile hedefine başarılı şekilde ulaştırılmıştır. Uçtan uca gecikme değeri ise yaklaşık olarak 143 ms’dir. Diğer senaryolarda olduğu gibi, en verimli PDR ve uçtan uca gecikme değerlerine sahip olan protokol VNPER olmuştur. Aynı metrikler dikkate alındığında VNPER’den sonra en verimli protokol BMFR olmuştur. Geriye kalan diğer protokoller arasında en verimli protokoller sırası ile CAR, MURU, ASTAR olmuştur. En fazla yönlendirme yüküne sahip olan protokol MURU olurken, en düşük yönlendirme yüküne sahip olan protokol ASTAR olmuştur. VNPER ve CAR protokolü MURU’dan sonra en fazla yönlendirme yüküne sahip olan protokollerdir. BMFR protokolü ise, en az yönlendirme yüküne sahip ikinci protokol olmuştur. Araç sayısı ve veri trafiğı miktarı arttıkça, VNPER protokolü ile diğer protokoller arasında yönlendirme yükü oranı farkı azalmaktadır. VANET ortamının yoğunlaşmasının yönlendirme yükü üzerindeki performansına olumsuz etkisi VNPER protokolünde daha az hissedilmektedir.

Çizelge 4.23. Üçüncü model üçüncü senaryodan elde edilen deneysel sonuçlar

Protokoller	Benzetim Süresi (sn)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	81 – 120	2237	88,62	143,45	24,05	53780	62,13
ASTAR	81 – 120	1710	67,74	169,56	20,43	34935	47,50
MURU	81 – 120	1776	70,36	159,45	28,14	49976	49,33
CAR	81 – 120	1908	75,59	154,12	24,78	47280	52,96
BMFR	81 – 120	2026	80,27	150,23	22,86	46314	56,27

Dördüncü senaryoda, 160 saniyeye kadar hem araç hem de veri trafiğı üretilmiştir ve senaryo 4 olarak isimlendirilmiştir. Tüm protokoller için aynı benzetim süresi aralığında aynı miktarda veri ve araç trafiğı üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.24’de sunulmuştur. Protokoller, aynı senaryo üzerinde çalıştırıldığı için aynı sayıda veri paketi ve araç sayısı üretilmiştir. 160 saniye boyunca yaklaşık 3268 adet veri paketi ve 341 adet araç üretilmiştir. Üretilen 3268 adet veri paketinden 2707 adeti VNPER protokolü ile hedefine başarılı şekilde ulaştırılmıştır. Uçtan uca gecikme değeri ise yaklaşık olarak 138 ms’dir. PDR E2E gecikme değeri en verimli protokol VNPER olmuştur. E2E gecime değeri ASTAR protokolünde en fazladır. BMFR protokolü en verimli ikinci protokol olmuştur. En fazla yönlendirme yüküne sahip olan protokol MURU olurken en az yönlendirme yüküne sahip olan protokol ise ASTAR olmuştur. VNPER, BMFR ve CAR protokolleri birbirlerine yakın oranlarda yönlendirme yüküne sahiptir. Ancak, VNPER protokolü, BMFR ve CAR protokollerine göre, daha yüksek oranda PDR oranına ve daha düşük oranda E2E gecikme oranına sahiptir.

Çizelge 4.24. Üçüncü model dördüncü senaryodan elde edilen deneysel sonuçlar

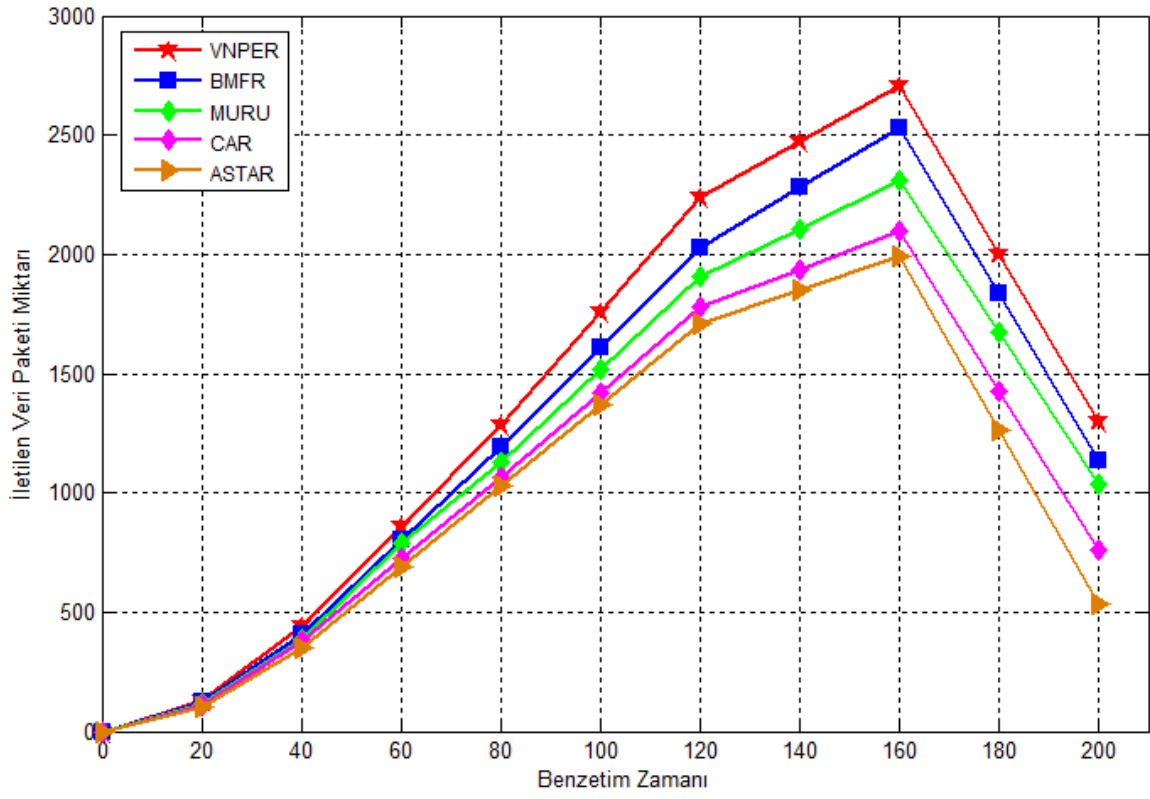
Protokoller	Benzetim Süresi (sn)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	121 – 160	2707	82,81	137,56	29,46	79748	57,26
ASTAR	121 – 160	1992	60,94	157,95	25,32	50437	42,72
MURU	121 – 160	2097	64,15	149,86	35,65	74758	45,08
CAR	121 – 160	2308	70,60	145,34	30,86	71224	48,62
BMFR	121 – 160	2531	77,42	142,67	29,42	74462	55,36

Beşinci senaryoda, 200 saniyeye kadar hem araç hem de veri trafiğı üretilmiştir ve senaryo 5 olarak isimlendirilmiştir. Tüm protokoller için aynı benzetim süresi aralığında aynı miktarda veri ve araç trafiğı üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği düğümler aynı tutulmuştur. Elde edilen sonuçlar Çizelge 4.25’de sunulmuştur. Protokoller, aynı senaryo üzerinde çalıştırıldığı için aynı sayıda veri paketi ve araç sayısı üretilmiştir. 200 saniye boyunca yaklaşık 4037 adet veri paketi ve 435 adet araç üretilmiştir. Üretilen 4037 adet veri paketinden 1297 adeti VNPER protokolü ile hedefine başarılı şekilde ulaştırılmıştır. Uçtan uca gecikme değeri ise yaklaşık olarak 198 ms’dir. PDR ve E2E gecikme değeri en başarılı olan protokol VNPER olmuştur. Aynı metrikler kıyaslandığında en verimli protokoller sırasıya, BMFR, CAR, MURU ve ASTAR protokolleridir. En düşük yönlendirme yüküne sahip olan protokol ise ASTAR’dır. ASTAR’dan sonra en düşük yönlendirme yüküne sahip olan protokol VNPER olmuştur. En fazla yönlendirme yüküne sahip olan protokol ise MURU olmuştur. VANET ortamı yoğunlaştıkça, VNPER protokolünün yönlendirme yükü verimliliğı ASTAR protokolü hariç diğer protokollerin önüne geçmiştir. ASTAR protokolünün PDR ve E2E gecikme verimliliğı ise, VNPER protokolünün gerisinde kalmıştır.

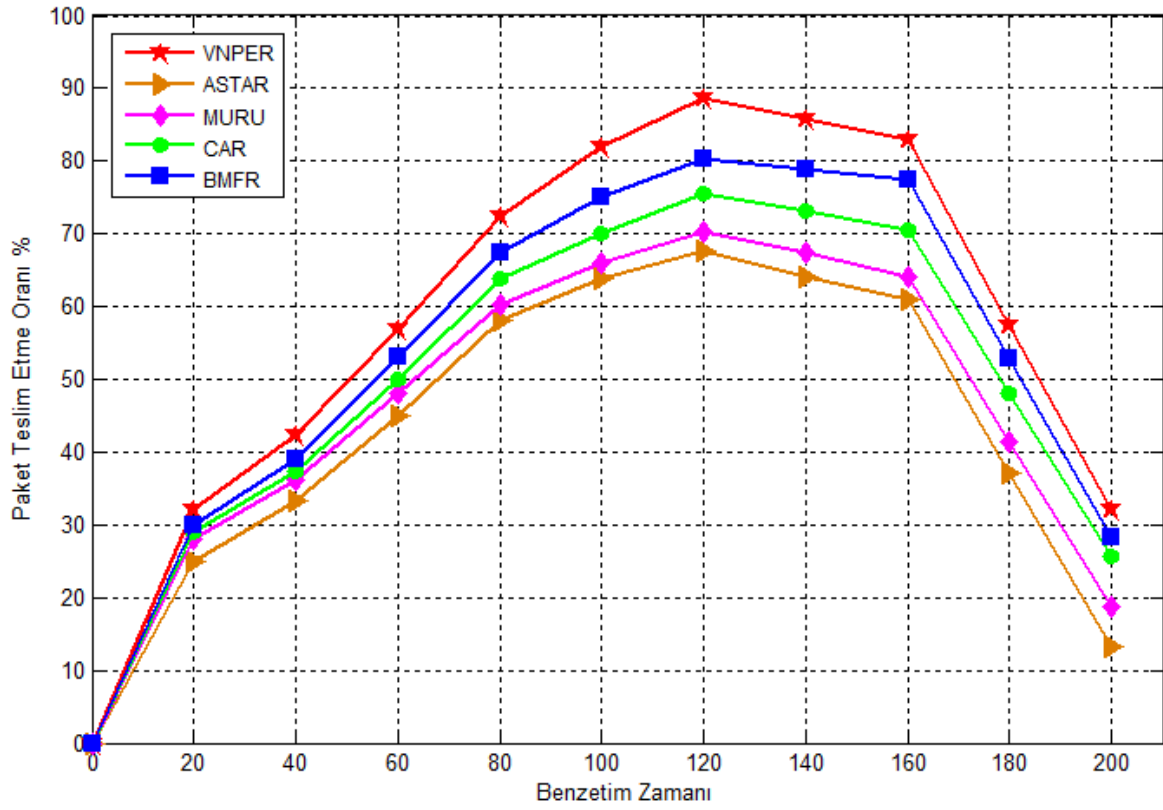
Çizelge 4.25. Üçüncü model beşinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Benzetim Süresi (sn)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	161 – 200	1297	32,12	197,36	51,14	66328	21,81
ASTAR	161 – 200	531	13,15	267,92	45,56	24192	9,15
MURU	161 – 200	758	18,78	249,57	73,15	55447	13,26
CAR	161 – 200	1035	25,64	230,78	63,93	66167	17,73
BMFR	161 – 200	1139	28,21	217,65	61,62	70185	19,97

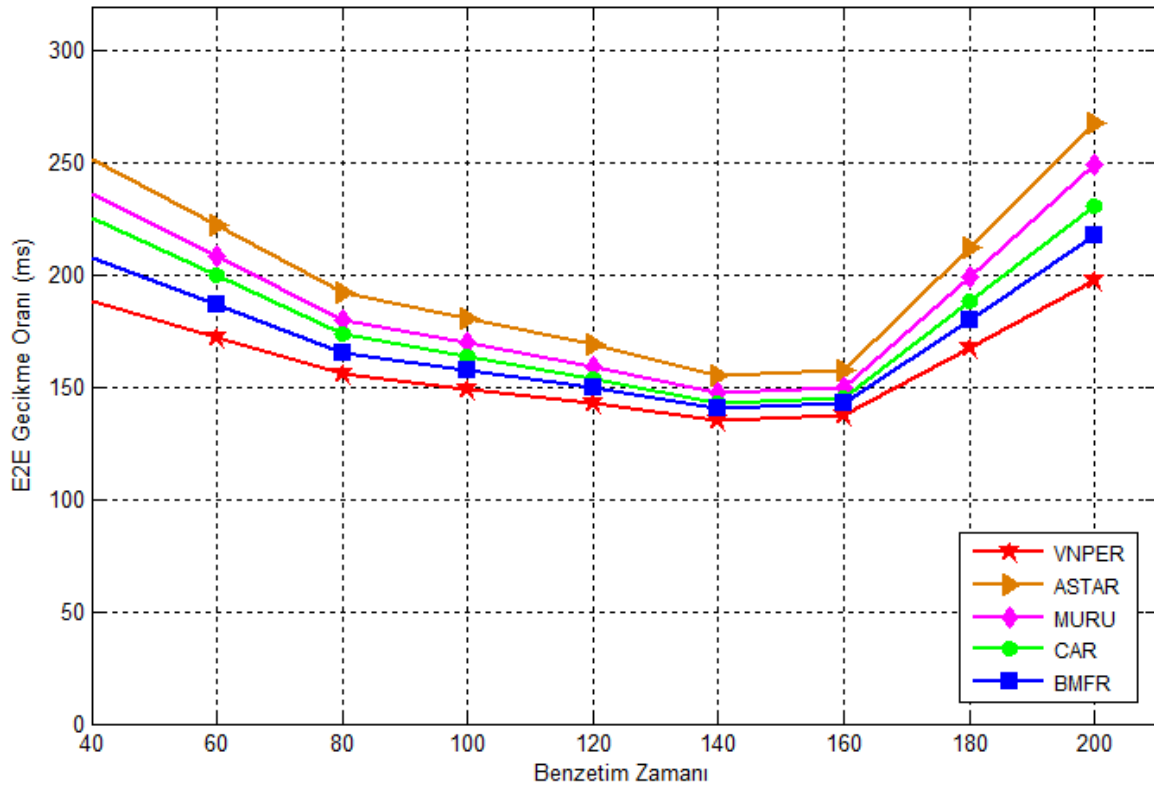
Deneysel sonuçlardan anlaşılacağı gibi, PDR ve E2E gecikme değeri en başarılı protokol VNPER olmuştur. VNPER protokolünden sonra aynı metriklere göre en başarılı ikinci protokol BMFR olmuştur. İlk senaryolarda en fazla yönlendirme yüküne sahip olan protokol VNPER olmasına rağmen, PDR ve E2E gecikme değeri en verimli olan protokoldür. VANET ortamının yoğunluğu arttıkça, VNPER protokolünün yönlendirme yükü verimliliği ASTAR protokolü hariç diğer protokollerin önüne geçmiştir. Özetle, VNPER protokolü ilk senaryolar için diğer protokollere göre masraflı olmasına rağmen en verimli protokoldür. Ayrıca, üretilen trafik miktarına ve araç sayısına göre, yönlendirme yükü en az değişkenlik gösteren protokol VNPER'dir. Üretilen trafik miktarı arttıkça, yönlendirme yükünde de artış olmaktadır. Ancak, en az artış oranı VNPER protokolündedir. Bundan dolayı, trafik yükü ve araç sayısı arttıkça VNPER protokolünün yönlendirme yükü diğer protokollere göre daha az olmaktadır. Şekil 4.37, 4.38, 4.39, 4.40, 4.41 ve 4.42'de üçüncü modelden elde edilen deneysel sonuçların farklı metriklere göre kıyaslanması sonucu ortaya çıkan grafiksel sonuçlar gösterilmiştir. Şekil 4.37'de kaynaktan hedefe başarılı şekilde ulaştırılan veri paketi miktarının kıyaslanması gösterilmiştir. Şekil 4.38'de protokollerin kaynaktan hedefe doğru paket teslim etme oranlarının kıyaslanması gösterilmiştir. Şekil 4.39'da bir veri paketinin kaynaktan hedefine doğru hangi oranda ortalama gecikme değeri ile ulaştığı karşılaştırmalı olarak gösterilmiştir. Şekil 4.40'da bir veri paketinin iletilmesi için üretilmesi gereken yönlendirme paketi oranı kıyaslamalı olarak gösterilmiştir. Şekil 4.41'de benzetim süresi boyunca üretilen toplam yönlendirme paketi sayıları karşılaştırmalı olarak gösterilmiştir. Şekil 4.42'de benzetim süresi boyunca protokollerin toplam işlem yapabilme kapasitesi karşılaştırmalı olarak gösterilmiştir.



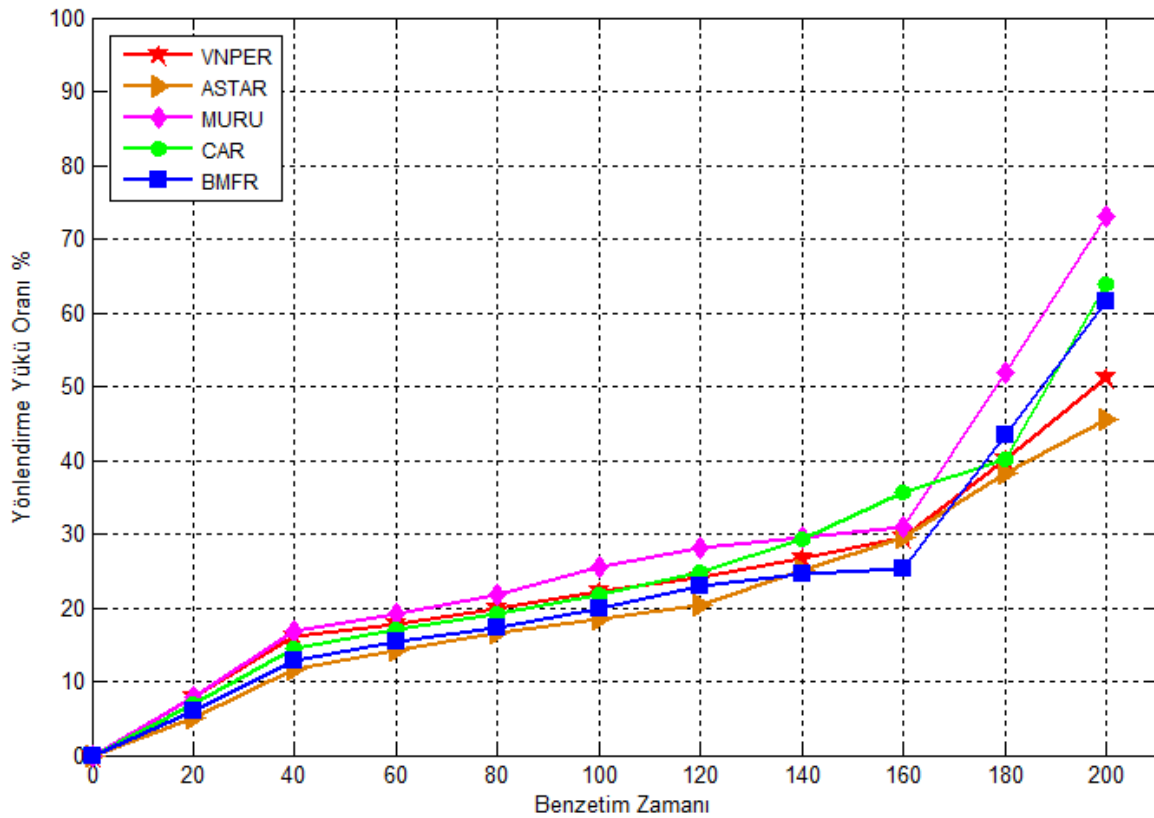
Şekil 4.37. Üçüncü model için iletilen veri paketi miktarı kıyaslanması



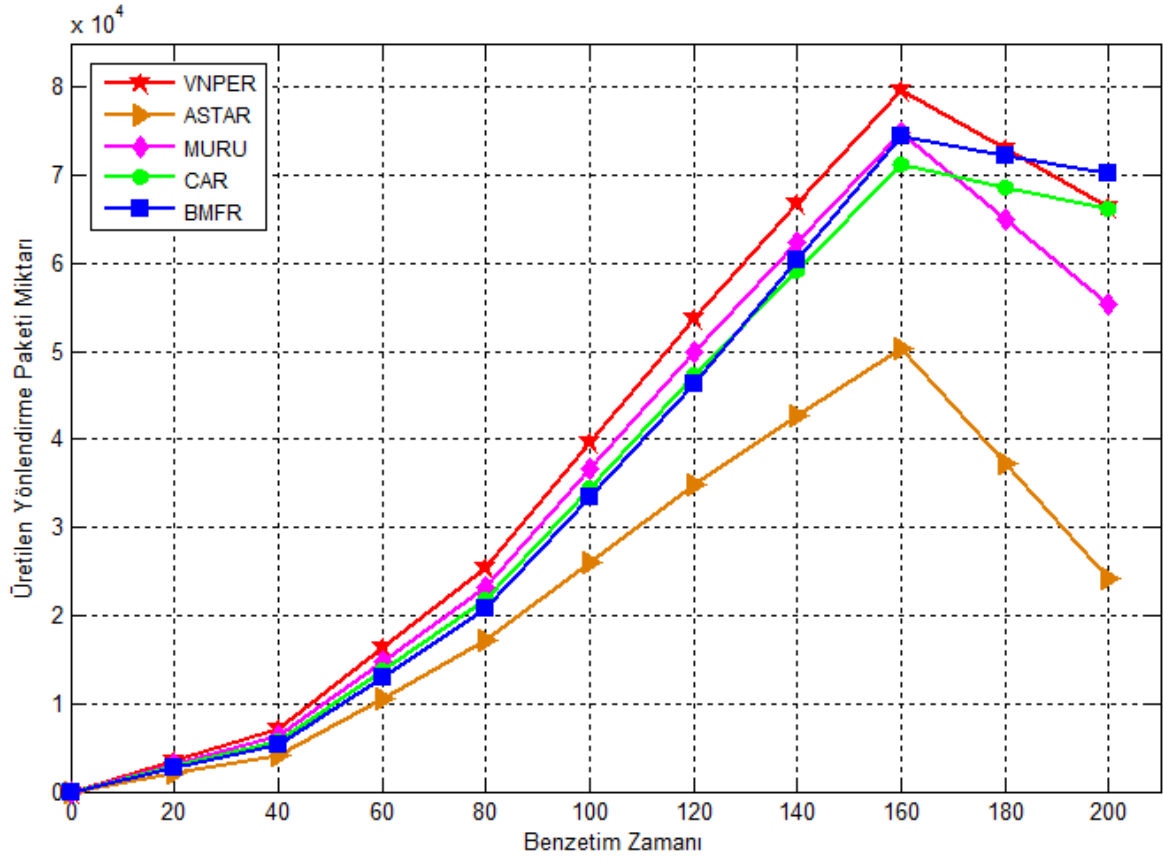
Şekil 4.38. Üçüncü model için paket teslim etme oranının kıyaslanması



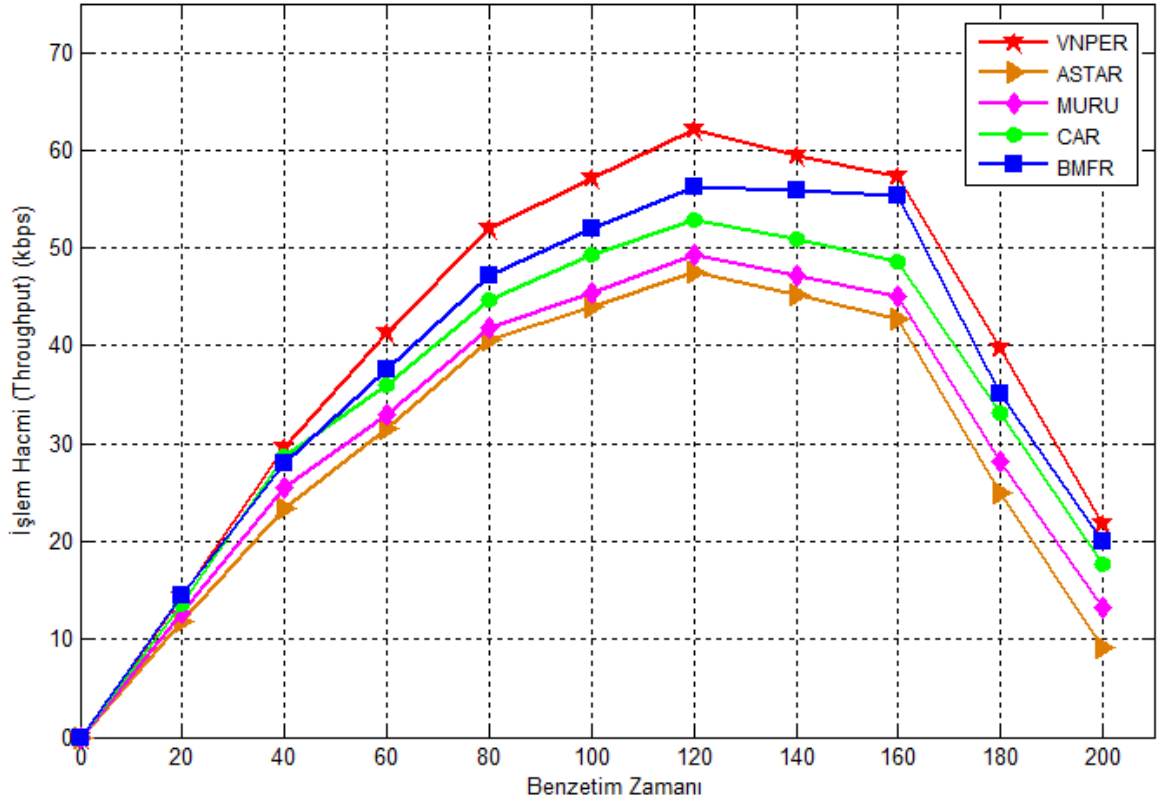
Şekil 4.39. Üçüncü model için uçtan uca gecikme oranının kıyaslanması



Şekil 4.40. Üçüncü model için yönlendirme yükü oranının kıyaslanması



Şekil 4.41. Üçüncü model için üretilen yönlendirme paketi miktarının kıyaslanması



Şekil 4.42. Üçüncü model için işlem hacmi miktarının kıyaslanması

4.4.4. Dördüncü modelden elde edilen deneysel sonuçlar

Dördüncü benzetim modelinde, 200 adet araç farklı noktalardan başlayarak farklı yönlere doğru hareket edecek şekilde 300 saniye boyunca araç trafiği oluşturulmuştur. Benzetim süresi boyunca toplamda yaklaşık 4000 adet veri paketi üretilmiştir. Üretilen veri trafik miktarı ve araç sayısı sabit tutulmuştur. Ancak, veri transferi yapacak olan kaynak ve hedef düğüm sayısı artırılmıştır. Veri transferinin gerçekleştiği kaynak ve hedef düğüm ikilisi PEER olarak isimlendirilmiştir. Farklı sayıda araç farklı noktalardan başlayarak farklı yönlere doğru hareket edecek şekilde 300 saniye boyunca araç trafiği oluşturulmuştur. Araçlar arasında ise veri trafiğinin oluşturulup kaynak araçtan hedef araca doğru iletilmesi sağlanmıştır. Test ortamında kullanılan NS2 parametreleri Çizelge 4.26’da gösterilmiştir. Oluşturulan veri trafiğinin hedef düğüme ne oranda ulaştığı, ne kadar sürede ulaştığı gibi ölçümler yapılarak protokollerin verimliliği test edilmiştir.

Çizelge 4.26. Dördüncü modelde kullanılan parametreler

Üretilen Veri Paketi Sayısı	4006 paket
Araç Sayısı	200 adet
Kanal Tipi	WirelessChannel
Yayılm Modeli	TwoRayGround
Interface Tipi	WirelessPhy
MAC Tipi	802_11
Kuyruk Tipi	DropTail/PriQueue
Anten tipi	OmniAntenna
Bağlantı Katmanı Tipi	LL
Kuyruk uzunluğu	50 paket
X Koordinat uzunluğu	16559 birim
Y Koordinat uzunluğu	6656 birim
Benzetim süresi	300 sn

Test haritası üzerinde bulunan 6 tane farklı noktadan farklı hedeflere doğru araç trafiği başlatılmıştır. Örneğin, Kurtuluş kavşağından Gazi Mustafa Kemal Bulvarına doğru 0. saniyeden 300. saniyeye kadar 40 adet araç oluşturulmuştur. Test ortamında oluşturulan araç trafik hareketleri Çizelge 4.27’de gösterilmiştir. Araç trafiğinin sumo ortamında oluşturulması ise Şekil 4.43’de gösterilmiştir.

Çizelge 4.27. Dördüncü modelde üretilen araç sayısı ve trafik akışı

Başlangıç Noktası	Bitiş Noktası	Başlangıç Süresi	Bitiş Süresi	Araç Sayısı
Kurtuluş Kavşağı	G.M.K bulvarı	0	300	40
Kurtuluş Kavşağı	Altınsoy Cad.	0	300	30
Atatürk Bulvarı	Kurtuluş Kavşağı	0	300	30
Strazburg Cad.	Necatibey Cad.	0	300	30
Atatürk Bulvarı	Fevzi Çakmak Sok.	0	300	30
Atatürk Bulvarı	G.M.K bulvarı	0	300	40

```

<flows>
<flow id="flow1" from="12195038#2" to="68695137#5" begin="0" end="300" no="40" />
<flow id="flow2" from="12195038#0" to="28470416#6" begin="0" end="300" no="30" />
<flow id="flow3" from="68695134#3" to="12195038#0" begin="0" end="300" no="30" />
<flow id="flow4" from="26302861#0" to="26317869#4" begin="0" end="300" no="30" />
<flow id="flow5" from="267158069#0" to="46039706#2" begin="0" end="300" no="30" />
<flow id="flow6" from="267158069#1" to="68695130#4" begin="0" end="300" no="40" />
</flows>

```

Şekil 4.43. Dördüncü model için araç trafiğinin sumo ortamında oluşturulması

İlk senaryoda, 200 adet araç farklı noktalardan başlayarak farklı yönlere doğru hareket edecek şekilde 300 saniye boyunca araç trafiği oluşturulmuştur. Benzetim süresi boyunca toplamda yaklaşık 4000 adet paketten oluşan bir veri trafiği üretilmiştir. Veri transferini gerçekleştirecek olan peer sayısı 1 olarak belirlenmiş ve senaryo 1 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği ve araç miktarı üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği peer düğümlerin aynı olması sağlanmıştır. Elde edilen sonuçlar Çizelge 4.28'de sunulmuştur. Protokoller, aynı senaryo üzerinde çalıştırıldığı için aynı sayıda veri paketi üretilmiştir. Üretilen 4006 adet veri paketinden 3066 adeti VNPER protokolü ile hedefine başarılı şekilde ulaştırılmıştır. Paket teslim etme oranı yaklaşık % 77 ile en başarılı protokol VNPER olmuştur. BMFR protokolü yaklaşık % 69 paket teslim etme oranı ile en başarılı ikinci protokol olmuştur. E2E gecikme değeri incelendiğinde, VNPER ve BMFR protokolleri yaklaşık 192 ms ile birbirlerine yakın sonuçlar elde etmiştir. Yönlendirme yükü en az olan protokol ASTAR olurken, en fazla yönlendirme yüküne sahip olan protokol MURU olmuştur. ASTAR'dan sonra en az yönlendirme yüküne sahip olan protokol VNPER olmuştur. VNPER protokolü yaklaşık % 23 yönlendirme yükü ile yaklaşık olarak 71653 adet yönlendirme paketi üretmiştir. VNPER'den sonra en verimli protokol olan BMFR protokolü ise yaklaşık % 27 yönlendirme yükü ile yaklaşık olarak 74410 adet yönlendirme paketi üretmiştir.

Çizelge 4.28. Dördüncü model birinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Düğüm Çifti Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	1	3066	76,53	192,51	23,37	71653	53,46
ASTAR	1	2219	55,39	243,95	19,27	42760	38,83
MURU	1	2446	61,06	232,96	32,48	79446	42,80
CAR	1	2531	63,18	219,44	24,79	62744	42,87
BMFR	1	2760	68,90	192,49	26,96	74410	49,69

İkinci senaryoda, 200 adet araç farklı noktalardan başlayarak farklı yönlere doğru hareket edecek şekilde 300 saniye boyunca araç trafiği oluşturulmuştur. Benzetim süresi boyunca toplamda 4006 adet paketten oluşan bir veri trafiği üretilmiştir. Veri transferini gerçekleştirecek olan peer sayısı 3 olarak belirlenmiş ve senaryo 2 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği ve araç miktarı üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği peer düğümlerin aynı olması sağlanmıştır. Elde edilen sonuçlar Çizelge 4.29'da sunulmuştur. PDR oranı yaklaşık % 76 ile en başarılı protokol VNPER olmuştur. VNPER'e en yakın PDR oranına sahip olan protokol yaklaşık % 68 ile BMFR protokolü olmuştur. En düşük E2E gecikme değerine sahip protokoller ise VNPER ve BMFR olmuştur. Her iki protokolde yaklaşık olarak 202 ms E2E gecikme değerine sahiptir. En yüksek gecikme değerine sahip olan protokoller ise sırasıyla ASTAR ve MURU protokolleri olmuştur. Yönlendirme yükü incelendiğinde, en fazla yönlendirme yüküne sahip olan protokol MURU olurken, en az yönlendirme yüküne sahip olan protokol ASTAR olmuştur. VNPER protokolü ASTAR protokolünden sonra en az yönlendirme yüküne sahip protokol olmuştur.

Çizelge 4.29. Dördüncü model ikinci senaryodan elde edilen deneysel sonuçlar

Protokoller	Düğüm Çifti Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	3	3025	75,51	202,57	26,41	79890	141,71
ASTAR	3	2161	53,94	248,62	21,35	46137	102,02
MURU	3	2389	59,64	243,58	37,89	90519	112,73
CAR	3	2482	61,96	232,63	28,23	70067	116,14
BMFR	3	2715	67,77	202,87	31,12	84491	127,10

Üçüncü senaryoda, 200 adet araç farklı noktalardan başlayarak farklı yönlere doğru hareket edecek şekilde 300 saniye boyunca araç trafiği oluşturulmuştur. Benzetim süresi boyunca toplamda 4006 adet paketten oluşan bir veri trafiği üretilmiştir. Veri transferini gerçekleştirecek olan peer sayısı 5 olarak belirlenmiş ve senaryo 3 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği ve araç miktarı üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği peer düğümlerin aynı olması sağlanmıştır. Elde edilen sonuçlar Çizelge 4.30'da sunulmuştur. Diğer senaryolarda olduğu gibi, en çok PDR değerine sahip olan protokol VNPER olmuştur. En düşük E2E gecikme değerine sahip olan protokol BMFR olmuştur. Ancak, VNPER ile BMFR protokolleri arasındaki PDR oranı farkı, E2E gecikme değeri arasındaki farktan daha fazladır. Bu açıdan değerlendirildiğinde, VNPER protokolü BMFR protokolüne göre daha başarılıdır. Geriye kalan diğer protokoller arasında en düşük E2E gecikme değerine sahip olan protokoller sırası ile CAR, MURU ve ASTAR olmuştur. En fazla yönlendirme yüküne sahip olan protokol MURU olurken, en düşük yönlendirme yüküne sahip olan protokol ise ASTAR olmuştur. ASTAR'dan sonra en az yönlendirme yüküne sahip olan protokoller sırasıyla VNPER, CAR ve BMFR olmuştur.

Çizelge 4.30. Dördüncü model üçüncü senaryodan elde edilen deneysel sonuçlar

Protokoller	Düğüm Çifti Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	5	2926	73,04	217,85	31,16	91174	137,17
ASTAR	5	1975	49,30	267,75	24,47	48328	93,17
MURU	5	2206	55,07	259,88	45,72	100858	106,36
CAR	5	2331	58,19	249,45	33,68	78508	110,43
BMFR	5	2579	64,38	216,21	37,21	95965	121,31

Dördüncü senaryoda, 200 adet araç farklı noktalardan başlayarak farklı yönler doğru hareket edecek şekilde 300 saniye boyunca araç trafiğı oluşturulmuştur. Benzetim süresi boyunca toplamda 4006 adet paketten oluşan bir veri trafiğı üretilmiştir. Veri transferini gerçekleştirecek olan peer sayısı 10 olarak belirlenmiş ve senaryo 4 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiğı ve araç miktarı üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği peer düğümlerin aynı olması sağlanmıştır. Elde edilen sonuçlar Çizelge 4.31’de sunulmuştur. PDR ve E2E gecikme değeri en verimli protokol VNPER olmuştur. E2E gecikme değeri incelendiğinde BMFR protokolü az bir fark ile VNPER’den sonra en verimli ikinci protokol olmuştur. En yüksek E2E gecikme değerine sahip olan protokol ASTAR olmuştur. En fazla yönlendirme yüküne sahip olan protokol MURU olurken en az yönlendirme yüküne sahip olan protokol ise ASTAR olmuştur. ASTAR’dan sonra en az yönlendirme yüküne sahip olan protokol VNPER olmuştur. VANET ortamındaki düğüm çifti sayısı arttıkça ASTAR protokolü ile VNPER protokolü arasındaki yönlendirme yükü farkı artmıştır. Buna ek olarak, VNPER protokolü ile diğer protokoller arasındaki yönlendirme yükü farkı da artmaktadır. Bunun sonucunda, ağdaki düğüm çifti sayısının artmasının olumsuz etkisi VNPER protokolünde diğer protokollere oranla daha az hissedilmektedir.

Çizelge 4.31. Dördüncü model dördüncü senaryodan elde edilen deneysel sonuçlar

Protokoller	Düğüm Çifti Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	10	2645	66,03	254,62	47,76	126328	58,64
ASTAR	10	1410	35,19	310,54	32,95	46460	31,55
MURU	10	1649	41,16	300,14	70,32	155958	36,56
CAR	10	1899	47,40	293,46	50,63	96146	42,11
BMFR	10	2213	55,24	255,97	56,84	125787	51,63

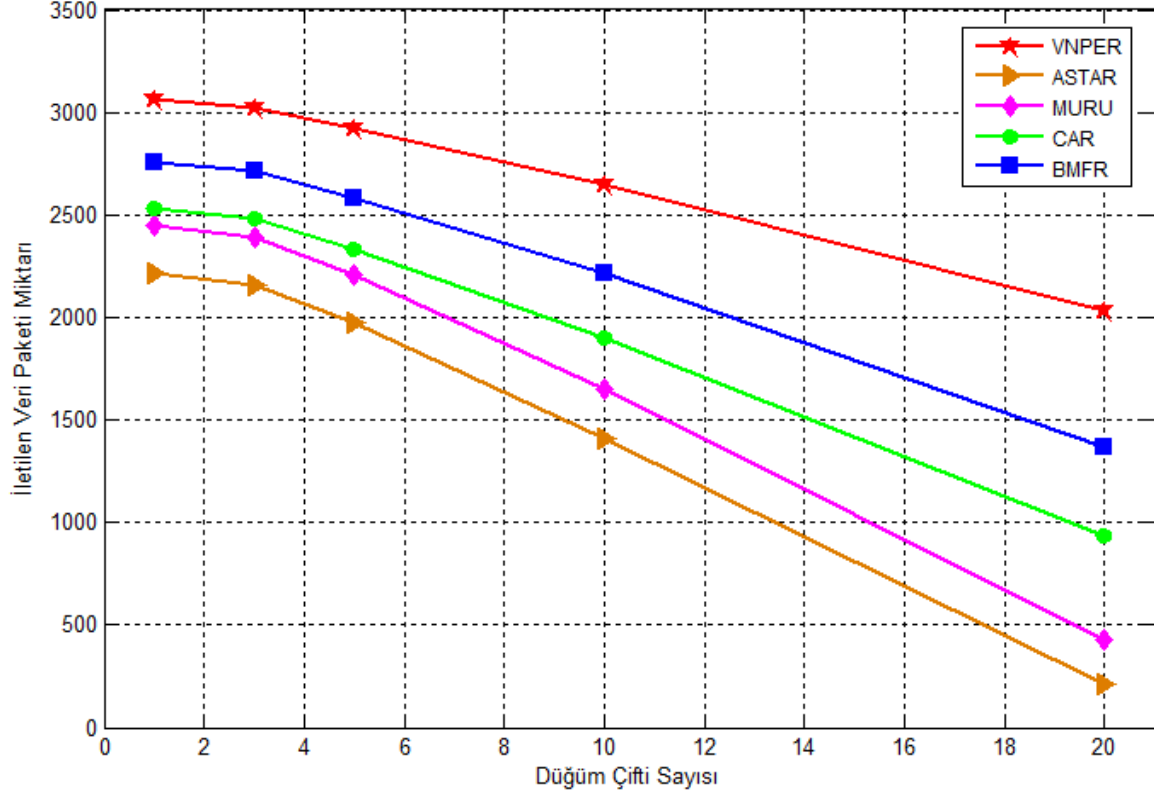
Beşinci senaryoda, 200 adet araç farklı noktalardan başlayarak farklı yönlere doğru hareket edecek şekilde 300 saniye boyunca araç trafiği oluşturulmuştur. Benzetim süresi boyunca toplamda 4006 adet paketten oluşan bir veri trafiği üretilmiştir. Veri transferini gerçekleştirecek olan peer sayısı 20 olarak belirlenmiş ve senaryo 5 olarak isimlendirilmiştir. Tüm protokoller için eşit miktarda veri trafiği ve araç miktarı üretilmiştir. Buna ek olarak, veri akışının gerçekleşeceği peer düğümlerin aynı olması sağlanmıştır. Elde edilen sonuçlar Çizelge 4.32’de sunulmuştur. PDR ve E2E gecikme değerleri en başarılı olan protokol VNPER olmuştur. VNPER’den sonra en düşük E2E değerine sahip protokol BMFR olurken, en yüksek E2E gecikme değerine sahip olan protokol yaklaşık 392 ms ile ASTAR olmuştur. En düşük yönlendirme yüküne sahip olan protokol ASTAR olurken, en yüksek yönlendirme yüküne sahip olan protokol ise yaklaşık % 98 ile MURU olmuştur. En fazla yönlendirme paketi VNPER protokolü tarafından üretilmiştir. Ancak, hedefine başarılı şekilde ulaştırılan veri paketi miktarı dikkate alındığında, yönlendirme paketinin çok olmasının bir önemi olmamaktadır.

Çizelge 4.32. Dördüncü model beşinci senaryodan elde edilen deneysel sonuçlar

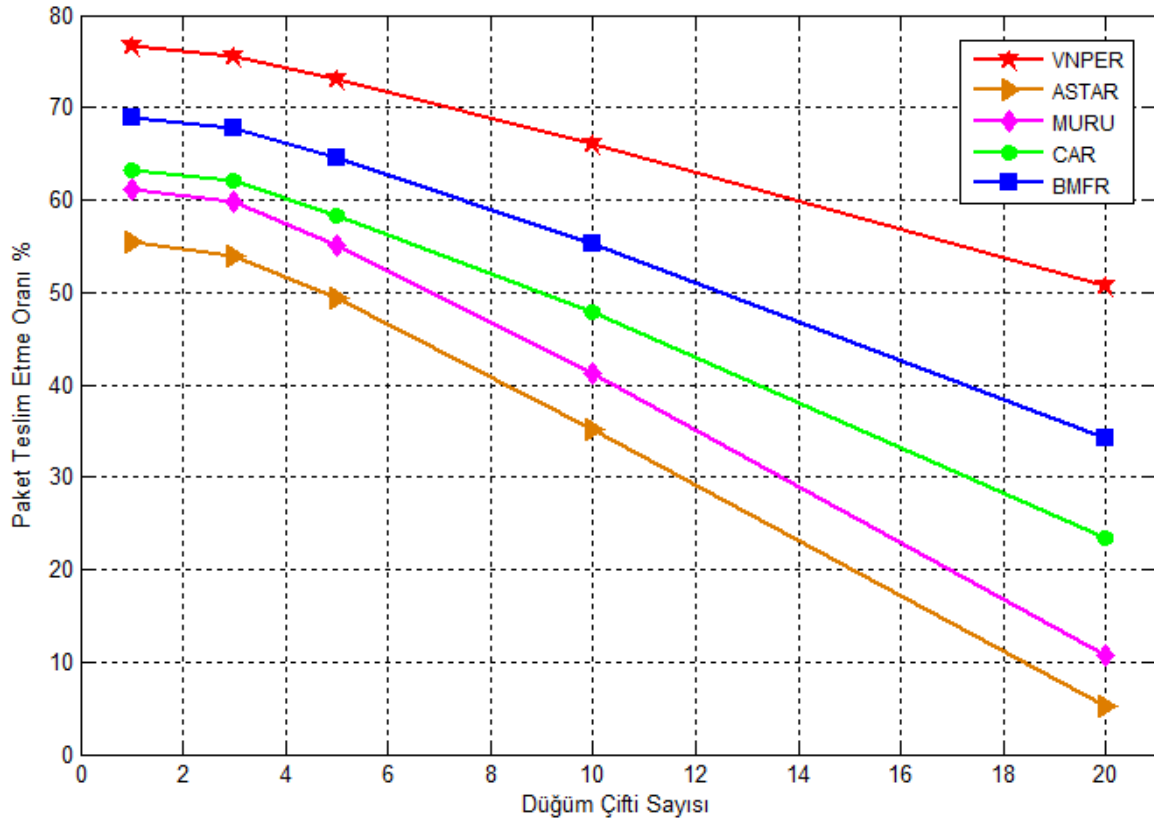
Protokoller	Düğüm Çifti Sayısı (Adet)	Alınan Paket (Adet)	PDR Oranı (%)	Uçtan Uca Gecikme (ms)	Yönlendirme Yüğü (%)	Yönlendirme Paketi (Adet)	Throughput (Kbps)
VNPER	20	2029	50,65	326,71	65,76	133427	32,76
ASTAR	20	211	5,26	392,35	46,63	9839	4,00
MURU	20	431	10,76	374,37	98,92	42635	6,90
CAR	20	937	23,39	373,74	70,72	66265	14,54
BMFR	20	1366	34,10	329,32	78,13	106726	22,47

Deneysel sonuçlardan anlaşılacağı gibi, PDR değeri en başarılı protokol VNPER olmuştur. E2E gecikme değeri incelendiğinde, VNPER ve BMFR protokolleri birbirlerine yakın sonuçlar elde etmiştir. Düğüm çifti sayısı arttıkça, VNPER protokolü BMFR protokolünden daha az E2E gecikme değerine sahip olmuştur. En düşük PDR oranına sahip olan protokol ASTAR olmuştur. VNPER protokolünden sonra en başarılı protokol BMFR olmuştur. Diğer protokoller arasında en başarılı protokoller sırasıyla, CAR, MURU ve ASTAR olmuştur. En fazla yönlendirme yükü oranına sahip olan protokol MURU olurken, en düşük yönlendirme yükü oranına sahip protokol ASTAR olmuştur. En fazla yönlendirme paketi üreten protokol VNPER olmasına rağmen, PDR ve E2E değeri en verimli olan protokoldür. Özetle, diğer protokollere göre, VNPER protokolü daha fazla yönlendirme paketi üretmesine rağmen en verimli protokoldür. Ayrıca, düğüm çifti sayısına göre, yönlendirme yükü en az değişkenlik gösteren protokol VNPER protokolüdür. Düğüm çifti sayısı arttıkça, yönlendirme yükünde de artış olmaktadır. Ancak, en az artış oranı VNPER protokolündedir. Bundan dolayı, düğüm çifti miktarı arttıkça VNPER protokolünün yönlendirme yükü diğer protokollere göre daha az olmaktadır. Şekil 4.44, 4.45, 4.46, 4.47, 4.48 ve 4.49’de üçüncü modelden elde edilen deneysel sonuçların farklı metriklere göre kıyaslanması sonucu ortaya çıkan grafiksel sonuçlar gösterilmiştir. Şekil 4.44’de kaynaktan hedefe başarılı şekilde ulaştırılan veri paketi miktarının kıyaslanması gösterilmiştir. Şekil 4.45’de protokollerin kaynaktan hedefe doğru paket teslim etme oranlarının kıyaslanması gösterilmiştir. Şekil 4.46’da bir veri paketinin kaynaktan hedefine doğru hangi oranda ortalama gecikme değeri ile ulaştığı karşılaştırmalı olarak gösterilmiştir. Şekil 4.47’de bir veri paketinin iletilmesi için üretilmesi gereken yönlendirme paketi oranı kıyaslamalı olarak gösterilmiştir. Şekil 4.48’de benzetim süresi boyunca üretilen toplam yönlendirme paketi sayıları karşılaştırmalı olarak gösterilmiştir.

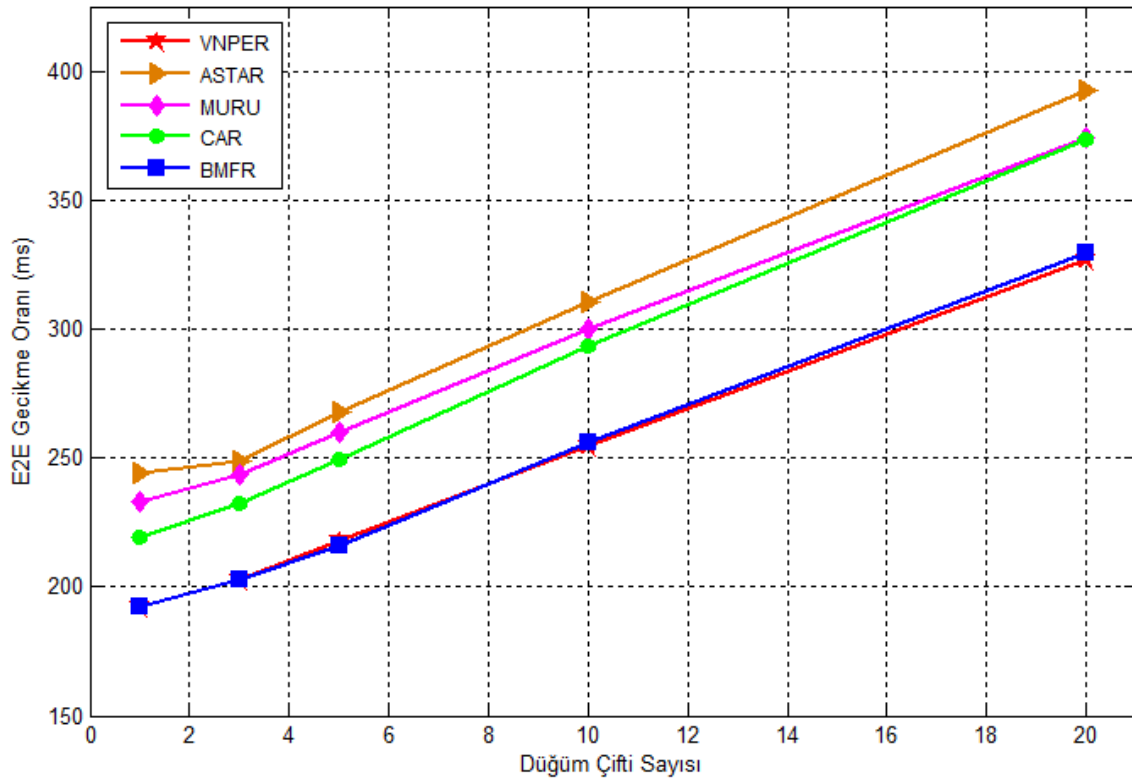
Şekil 4.49’de benzetim süresi boyunca protokollerin toplam işlem yapabilme kapasitesi karşılaştırmalı olarak gösterilmiştir.



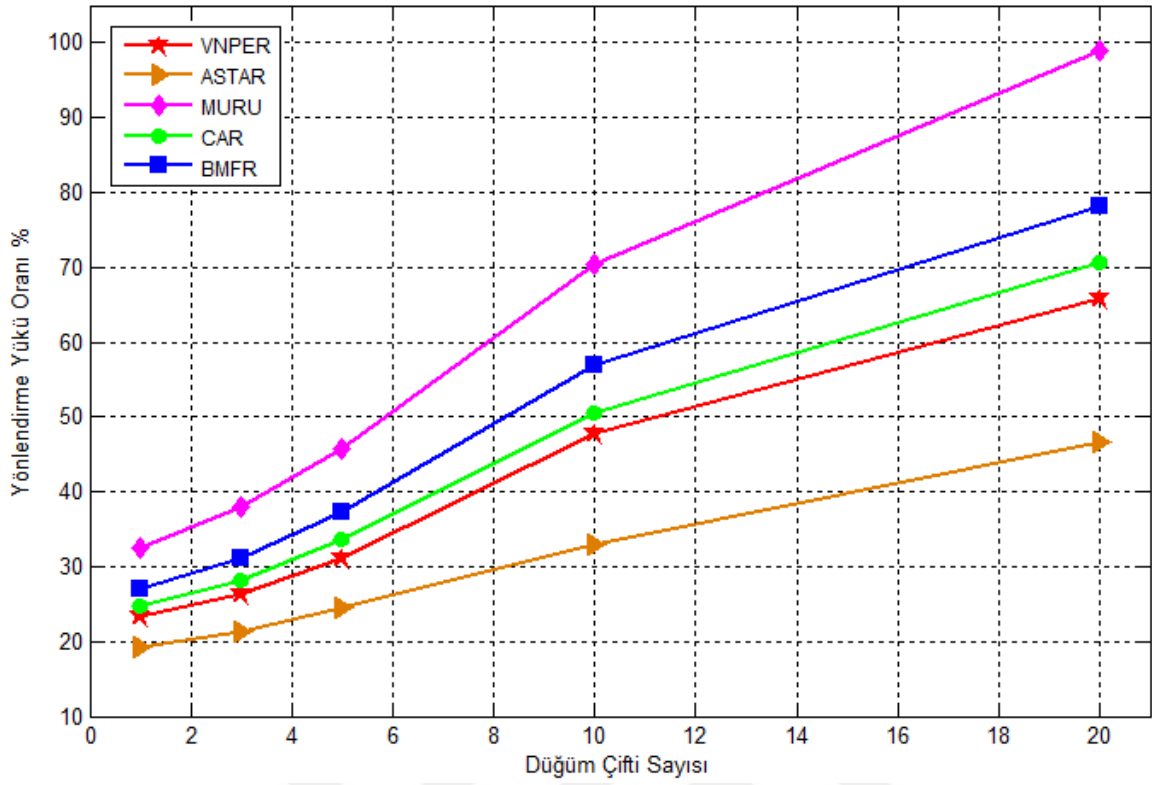
Şekil 4.44. Dördüncü model için iletilen veri paketi miktarının kıyaslanması



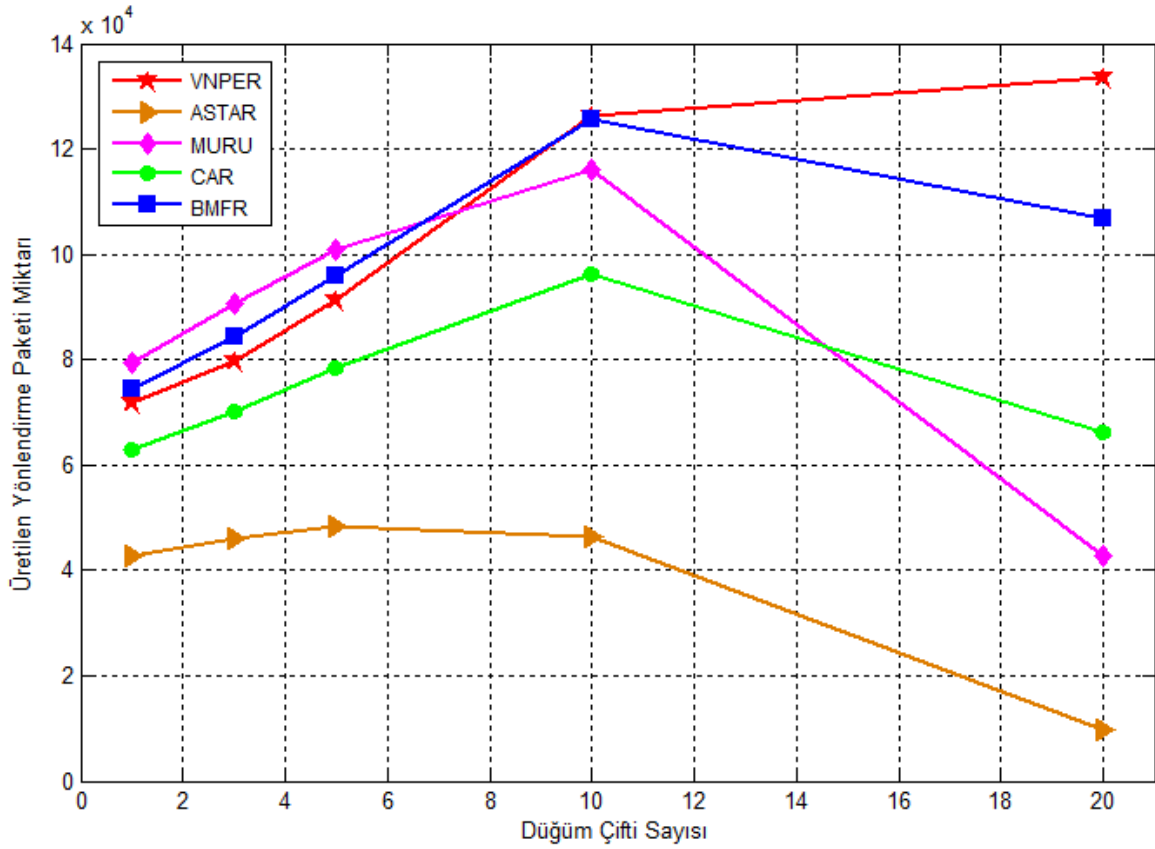
Şekil 4.45. Dördüncü model için paket teslim etme oranının kıyaslanması



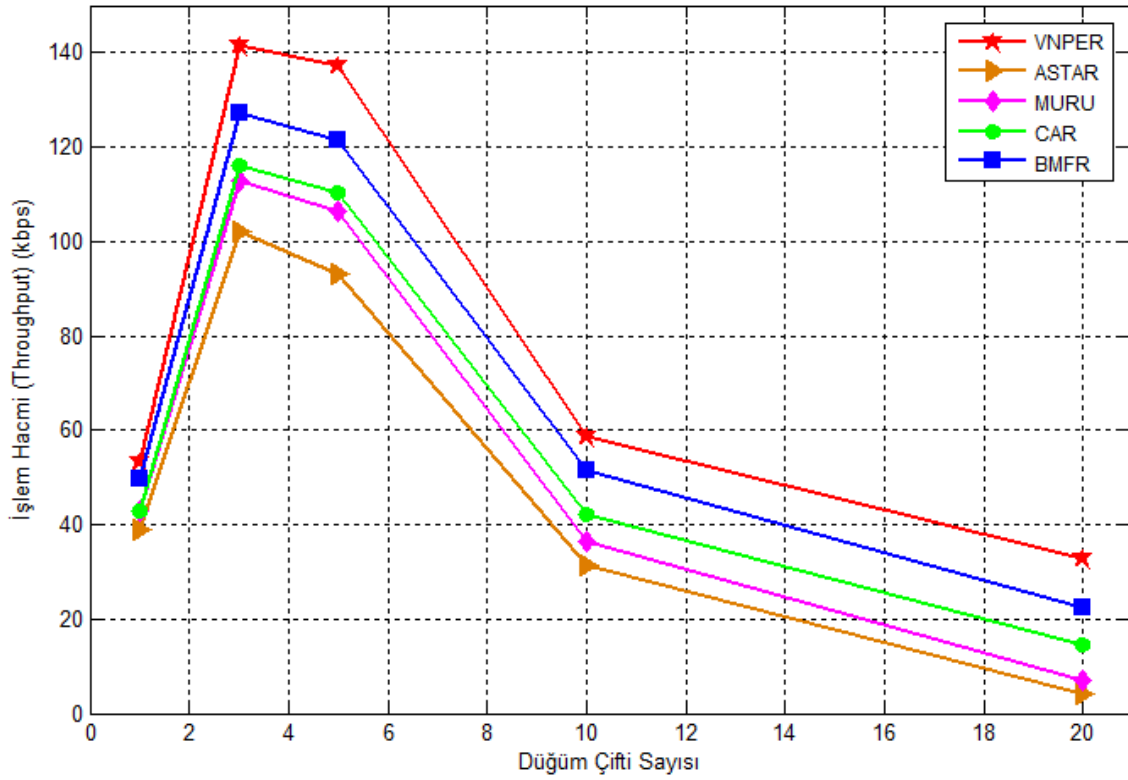
Şekil 4.46. Dördüncü model için uçtan uca gecikme oranının kıyaslanması



Şekil 4.47. Dördüncü model için yönlendirme yükü oranının kıyaslanması



Şekil 4.48. Dördüncü model için üretilen yönlendirme yükü miktarının kıyaslanması



Şekil 4.49. Dördüncü model için işlem hacmi miktarının kıyaslanması

Senaryolarda kullanılan araç sayısı, iletilen veri paketi miktarı ve düğüm çifti sayısı protokollerin verimliliğini ortaya çıkarabilmek için artırılmıştır. Elde edilen grafiksel sonuçlar incelendiğinde tüm model ve senaryolar için PDR, E2E ve toplam işlem hacmi metriklerine göre VNPER protokolü daha başarılı sonuçlar elde etmiştir. Tüm modellerdeki ilk senaryolarda VNPER protokolü diğer protokollere göre daha fazla yönlendirme yükü ile işlem yapmasına rağmen, araç sayısı, iletilen veri paketi miktarı ve düğüm çifti sayısında artış oldukça daha az yönlendirme yükü ile çalışmaktadır. Grafiksel sonuçlar incelendiğinde, tüm modellerdeki ilk iki senaryoda VNPER protokolü daha masraflı bir yönlendirme yüküne sahipken, geri kalan üç senaryoda daha masrafsız bir yönlendirme yüküne sahip olduğu görülmektedir. VNPER protokolünün diğer protokollere göre daha fazla ölçeklenebilirliğe sahip olduğu sonucuna ulaşılmaktadır.



5. SONUÇ VE ÖNERİLER

Bu tezde, araçsal ağlarda kullanılmak üzere bir yönlendirme protokolü geliştirilmiştir. Geliştirilen protokol VNPER olarak isimlendirilmiştir. Kentsel alanlarda kullanılmak üzere tasarlanan araçtan araca bir yönlendirme protokolüdür. Araçların kentsel alanlardaki hareketlerini dikkate alarak bir sonraki zaman diliminde araçların dizilimlerini tahmin etmeye dayalı sezgisel ve ölçeklenebilir bir yönlendirme protokolü olarak değerlendirilebilir. Araçsal ağlarda kullanım için geliştirilen protokoller farklı özelliklere göre sınıflandırılarak detaylı şekilde literatür taraması yapılmıştır. Uygulama ortamı için, NS2-Move-Sumo benzetim araçları birbirleri ile etkileşimli olarak çalışacak hale getirilmiştir. Kızılay-Sıhhiye meydanı, Kolej, Demirtepe, Maltepe, Tandoğan ve Necatibey bölgesi uygulama ortamı olarak belirlenmiştir. OSM üzerinden bu bölgenin gerçek dünya haritası alınmıştır. Netconverter ile linux platformunda çeviri işlemi yapılarak, Sumo ile çalışabilecek hale getirilmiştir. VNPER protokolünde kullanılmak amacıyla paket tipleri ve zamanlayıcılar geliştirilmiştir. İleri görüşlü komşuluk algoritması ve ileri görüşlü komşu keşif yaklaşımı olarak adlandırılan yeni yaklaşımlar öne sürülmüştür. Rota keşif maliyetini azaltmak için yedek rotaların tutulduğu topoloji tablosu olarak adlandırılan yeni bir tablo geliştirilmiştir. Düğümlerin paketleri aldıkları zaman çalışacağı algoritmalar belirlenerek akış şemaları çıkartılmıştır. Rotaların maliyetinin belirlenmesi için matematiksel bir model ortaya konulmuştur. Rotalar arasında seçim yapılacağı zaman dikkate alınacak özellikler matematiksel formül ile ifade edilmiştir.

V2V iletişim modeli için geliştirilen VNPER yönlendirme protokolünün C++ kodu ve TCL kodu yazılarak, NS2 ile bütünleşik hale getirilmiştir. NS2 ile bütünleşik hale getirilmesinden sonra hata ayıklama işlemleri gerçekleştirilmiştir. VNPER protokolü ile kıyaslanacak olan protokoller literatürden tespit edilmiştir. Protokollerin verimliliğini ölçebilmek için 4 tane metrik belirlenmiştir. Bu metriklerin kullanılabilmesi için awk betik dili kullanılarak betikler yazılmıştır. VNPER ve diğer protokoller belirlenen metriklere göre kıyaslaması yapılarak VNPER protokolünün verimliliği ortaya çıkartılmıştır. Aynı uygulama ortamı üzerinde 4 farklı benzetim modeli kullanılarak deneyler gerçekleştirilmiştir. Her bir model için 5 farklı senaryodan elde edilen deneysel sonuçlara göre, PDR, E2E ve toplam işlem hacmi metriklerine göre en başarılı protokol VNPER olmuştur. Yönlendirme yükü ilk senaryolar için en fazla VNPER olmasına rağmen veri trafiği miktarı, araç sayısı ya da düğüm çifti

sayısı arttıkça yönlendirme yükünde meydana gelen artış en az VNPER protokolünde olmuştur. VNPER protokolü modellerdeki ilk senaryolar için diğer protokollere göre masraflı olmasına rağmen PDR, E2E ve toplam işlem hacmi metriklerine göre en verimli protokoldür. Ayrıca, üretilen trafik miktarına göre, yönlendirme yükü en az değişkenlik gösteren protokol VNPER'dir. Üretilen trafik miktarı arttıkça, yönlendirme yükünde de artış olmaktadır. Ancak, en az artış oranı VNPER protokolündedir. Bundan dolayı, trafik yükü arttıkça VNPER protokolünün yönlendirme yükü diğer protokollere göre daha az olmaktadır. Bu durum göz önünde bulundurulduğunda, VNPER protokolünün ölçeklenebilirlik açısından en verimli protokol olduğu sonucuna ulaşılmaktadır.

Araçsal ağların etkin şekilde gelecek günlerde kullanılabilmesi olası saldırılara karşı korunaklı olmasına bağlıdır. Araçsal ağ ortamına yapılabilecek saldırıların ilerleyen zamanlarda giderek artması beklenmektedir. Özellikle araçsal ağlar için geliştirilen yönlendirme protokollerinin bu güvenlik gereksinimlerini karşılayabilecek hale gelmeleri gerekmektedir. İlerleyen zamanlarda güvenlik tabanlı yönlendirme protokollerinin geliştirilmesi hedeflenmektedir.

KAYNAKLAR

1. Campolo, C., Molinaro, A. (2011). *On vehicle-to-roadside communications in 802.11p/WAVE VANETs*. Wireless Communications and Networking Conference (WCNC), 1010-1015.
2. Teixeira, F. A., Silva, V. F., Leoni, J. L., Macedo, D. F., Nogueira, M. S. (2014). Vehicular networks using the IEEE 802.11p standard: An experimental analysis. *Vehicular Communications*, 1(2), 91-96.
3. Abdelgader, A. M. S., Lenan, W. (2014). *The Physical Layer of the IEEE 802.11p WAVE Communication Standard: The Specifications and Challenges*. Proceedings of the World Congress on Engineering and Computer Science, 691-698.
4. Qiu, H. J. F., Ho, I. W., Tse, C. K. (2015). A Methodology for Studying 802.11p VANET Broadcasting Performance with Practical Vehicle Distribution. *IEEE Transactions on Vehicular Technology*, 64(10), 4756-4769.
5. Silva, F. A., Boukerche, A., Silva, T. R. M. B., Ruiz, L. B., Loureiro, A. A. F. (2015). A novel macroscopic mobility model for vehicular networks. *Computer Networks*, 79, 188-202.
6. Wahab, O. A., Mourad, A., Otrok, H., Bentahar, J. (2016). CEAP: SWM-based intelligent detection model for clustered vehicular ad hoc networks. *Expert Systems with Applications*, 50, 40-54.
7. Shelly, S., Vijay, V., Babu, A. V. (2015). Model for Path Duration in Vehicular Ad Hoc Networks under Greedy Forwarding Strategy. *Procedia Computer Science*, 48, 394-400.
8. Arkian, H. R., Atani, R. E., Pourkhalili, A, Kamali, S. (2014). Cluster-based traffic information generalization in Vehicular Ad-hoc Networks. *Vehicular Communications*, 1(4), 197-207.
9. Darwish, T., Abu Bakar, K. (2015). Traffic density estimation in vehicular ad hoc networks: A review. *Ad Hoc Networks*, 24(A), 337-351.
10. Timpner, J., Wolf, L. (2016). Query-response geocast for vehicular crowd sensing. *Ad Hoc Networks*, 36(2), 435-449.
11. Felice, M. D., Cerqueira, E., Melo, A., Gerla, M., Cuomo, F., Baiocchi, A. (2015). A distributed beaconless routing protocol for real-time video dissemination in multimedia VANETs. *Computer Communications*, 58(1), 40-52.
12. Karp, B., Kung, H.T. (2000). *GPSR: greedy perimeter stateless routing for wireless networks*. Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, 243-254.

13. Alsaqour, R. A., Abdelhaq, M., S., Asukour, O. A. (2012). Effect of network parameters on neighbor wireless link breaks in GPSR protocol and enhancement using mobility prediction model. *EURASIP Journal on Wireless Communications and Networking*, 171, 1-15.
14. Naumov, V., Baumann, R., Gross, T. (2016). *An evaluation of inter-vehicle ad hoc networks based on realistic vehicular ad hoc networks*. Proceedings of the 7th ACM international symposium on mobile ad hoc networking and computing, 108-119.
15. Naumov, V., Gross, T. R. (2007). *Connectivity-aware routing (CAR) in vehicular ad-hoc networks*. Proceedings of the 26th IEEE international conference on computer communications, 1919-1927.
16. Raw R. S., Lobiyal, D. (2010). *B-MFR routing protocol for vehicular ad hoc networks*. Proceedings of the networking and information technology, 420-423.
17. Song, T., Xia, W., Shen, L. (2010). *A cluster-based directional routing protocol in VANET*. Proceedings of the 12th IEEE international conference on communication technology, 1172-1175.
18. Bachir, A., Benslimane, A. (2003). *A multicast protocol in ad hoc networks inter-vehicle geocast*. Proceedings of the 57th IEEE semiannual vehicular technology conference, 2456-2460.
19. Maihofer, C., Eberhardt, R. (2004). *Geocast in vehicular environments: caching and transmission range control for improved efficiency*. Proceedings of 2004 IEEE intelligent vehicles symposium, 951-956.
20. Kihl, M., Sichitiu, M., Ekeroth, T., Rozenberg, M. (2007). *Reliable geographical multicast routing in vehicular ad-hoc networks*. Proceedings of 5th international conference on wired/wireless internet communications, 315-325.
21. Zeadally, S., Hunt, R., Chen, Y. S., Irwan, A., Hassan, A. (2012). Vehicular ad hoc networks (VANETs): status, results, and challenges. *Telecommunication System*, 50(4), 217-241.
22. Lee, S. J., Su, W., Gerla, M. (2002). On-demand multicast routing protocol in multihop wireless mobile networks. *Mobile and Network Applications*, 7, 441-453.
23. Durrezi, M., Durrezi, A., Barolli, L. (2005). *Emergency broadcast protocol for inter-vehicle communications*. Proceedings of the 11th international conference on parallel and distributed systems, 402-406.
24. Maia, G., Aquino, A. L. L., Viana, A., Boukerche, A., Loureiro, A. F. F. (2012). *HyDi: a hybrid data dissemination protocol for highway scenarios in vehicular ad hoc networks*. Proceeding of the 2nd ACM international symposium on design and analysis of intelligent vehicular networks and applications, 115-122.
25. Ahmed, S., Kanere, S. S. (2006). *SKVR: scalable knowledge-based routing architecture for public transport networks*. Proceedings of the 3rd international workshop on vehicular ad hoc networks, 92-93.

26. Zhao, J., Cao, G. (2008). VADD: vehicle-assisted data delivery in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 57, 1910-1922.
27. Mo, Z., Zhu, H., Makki, K., Pissinou, N. (2006). *MURU: a multi-hop routing protocol for urban vehicular ad hoc networks*. Proceedings of the 2006 third annual international conference on mobile and ubiquitous systems, network and services, 1-8.
28. Lee, K. C., Harri, J. U., Gerla, M. (2007). *Enhanced perimeter routing for geographic forwarding protocols in urban vehicular scenarios*. Proceedings of 2007 IEEE globecom workshops, 1-10.
29. Tonguz, O., Wisitpongphan, N., Bai, F., Mudalige, P., Sadekar, V. (2007). *Broadcasting in VANET*. Proceedings of 2007 Mobile Networking for Vehicular Environments, 7-12.
30. Nekovee, M., Bogason, B. B. (2007). *Reliable and efficient information dissemination in intermittently connected vehicular ad hoc networks*. Proceedings of the 65th IEEE vehicular technology conference (VTC2007), 2486-2490.
31. Korkmaz, G., Ekici, E., Özgüner, F., Özgüner, Ü. (2004). *Urban multi-hop broadcast protocol for inter-vehicle communication systems*. Proceedings of the 1st ACM international workshop on vehicular ad hoc networks, 76-85.
32. Luo, Y., Zhang, W., Hu, Y. (2010). *A new cluster based routing protocol for VANET*. Proceedings of the 2010 international conference on networks security wireless communications and trusted computing, 176-180.
33. Lochert, C., Hartenstein, H., Tian, J., Fussler, H., Hermann, D., Mauve M. A. (2003). *A routing strategy for vehicular ad hoc networks in city environments*. Proceedings of the IEEE intelligent vehicles symposium, 156-161.
34. Camp, T., Boleng, J., Wilcox, L. (2002). *Location information services in mobile ad hoc networks*. Proceedings of IEEE international conference on communications, 3318-3324.
35. Perkins, C. E., Royer, E. M. (1999). *Ad hoc on-demand distance vector routing*. Proceedings of the 2nd IEEE workshop on mobile computing systems and applications WMCSA'99, 90-100.
36. Johnson, D. B., Maltz, D. A. (1996). Dynamic source routing in ad hoc wireless networks. *Mobile Computing*, 153-181.
37. Tian, J., Han, L., Rothermel, K. (2003). *Spatially aware packet routing for mobile ad hoc inter-vehicle radio networks*. Proceedings of the 2003 IEEE on intelligent transportation systems, 1546-1551.
38. Lochert, C., Mauve, M., Fübler, H., Hartenstein, H. (2005). Geographic routing in city scenarios. *ACM SIGMOBILE Mobile Computing and Communications Review*, 9(1), 69-72.

39. Cha, S. H., Lee, K. W., Cho, H. S. (2012). Grid-based predictive geographic routing for inter-vehicle communication in urban areas. *International Journal of Distributed Sensor Network*, 3, 1-7.
40. Giudici, F., Pagani, E. (2005). Spatial and traffic-aware routing (STAR) for vehicular systems. *High Performance Computing and Communications*, 3726, 77-86.
41. Seet, B.C., Liu, G., Lee, B.S., Foh, C.H., Wong, K.J., Lee, K.K. (2004). *A-STAR: a mobile and ad hoc routing strategy for metropolis vehicular communications*. Network technologies, services and protocols, performance of computer and communication networks, mobile and wireless communications, Networking 2004, 989-999.
42. Tee, C., Lee, A. (2009). *Adaptive reactive routing for VANET in city environments*. Proceedings of the 10th international symposium on pervasive systems, algorithms and networks (ISPAN'09), 610-614.
43. Jerbi, M., Senouci, S. M., Meraihi, R., Ghamri-Doudane, Y. (2007). *An improved vehicular ad hoc routing protocol for city environments*. Proceedings of the IEEE international conference on communications, 3972-3979, 2007.
44. Wang, T., Wang, G. (2010). *TIBCRPH: traffic infrastructure based cluster routing protocol with handoff in VANET*, Proceedings of the 19th annual on wireless and optical communications conference, 1-5.
45. Nakom, N. N., Rojviboonchai, K. (2010). *DECA: density-aware reliable broadcasting in vehicular ad hoc networks*. Proceedings of 2010 international conference on electrical engineering/electronics computer telecommunications and information technology (ECTI/CON), 598-602.
46. Souza, A. B., Celestino, J., Xavier F. A., Oliveria, F. D., Patel, A., Latifi, M. (2013). *Stable multicast trees based on ant colony optimization for vehicular ad hoc networks*. Proceedings of international conference on information networking (ICOIN'13), 101-106.
47. Shafiee, K., Leung, V. C. M. (2011). Connectivity-aware minimum-delay geographic routing with vehicle tracking in VANETs. *Ad Hoc Networks*, 9, 131-141.
48. Bitam S., Mellouk, A., Zeadally, S. (2013). HyBR: A Hybrid Bio-inspired Bee swarm routing protocol for safety applications in Vehicular Ad hoc Networks (VANETs). *Journal of Systems Architecture*, 59, 953-967.
49. Wang, S., Lin, Y. (2013). PassCAR: A passive clustering aided routing protocol for vehicular ad hoc networks. *Computer Communications*, 36, 170-179.
50. Bernsen, J., Manivannan, D. (2012). RIVER: A reliable inter-vehicular routing protocol for vehicular ad hoc networks. *Computer Networks*, 56, 3795-3807.

51. Filho, J. G., Patel, A., Batista, B. L. A., Junior, J. C. (2016). A systematic technical survey of DTN and VDTN routing protocols. *Computer Standards & Interfaces*, 48, 139-159.
52. Cunha, F., Villas, L., Boukerche, A., Maia, G., Viana, A., Mini, R. A. F., Loureiro, A. A. F. (2016). Data communication in VANETs: Protocols application and challenges. *Ad Hoc Networks*, 44, 90-103.
53. Ndashimye, E., Ray, S. K., Sarkar, N. I., Gutierrez, J. A. (2017). Vehicle-to-infrastructure communication over multi-tier heterogeneous networks: A survey. *Computer Networks*, 112, 144-166.
54. Gandotra, P., Jha, R. K., Jain, S. (2017). A survey on device-to-device (D2D) communication: Architecture and security issues. *Journal of Network and Computer Applications*, 78, 9-29.
55. Mir, Z. H., Filali, F. (2017). Large-scale simulations and performance evaluation of connected cars - A V2V communication perspective. *Simulation Modelling Practice and Theory*, 72, 88-103.
56. Said, A. M., Marot, M., Ibrahim, A. W., Afifi, H. (2016). Modeling interactive real-time applications in VANETs with performance evaluation. *Computer Networks*, 104, 66-78.
57. Dietzel, S., Gürtler, J., Frank, K. (2016). A resilient in-network aggregation mechanism for VANETs based on dissemination redundancy. *Ad Hoc Networks*, 37, 101-109.
58. Liu, X., Yan, G. (2016). Analytically modeling data dissemination in vehicular ad hoc networks. *Ad Hoc Networks*, 52, 17-27.
59. Turcanu, I., Salvo, P., Baiocchi, A., Cuomo, F. (2016). An integrated VANET-based data dissemination and collection protocol for complex urban scenarios. *Ad Hoc Networks*, 52, 28-38.
60. Hussain, R., Rezaeifar, Z., Lee, Y. H., Oh, H. (2015). Secure and privacy-aware traffic information as a service in VANET-based clouds. *Pervasive and Mobile Computing*, 24, 194-209.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : KOCAOĞLU, Ramazan
 Uyuşu : T.C.
 Doğum tarihi ve yeri : 28.06.1986, Ankara
 Medeni hali : Evli
 Telefon : 0 (312) 363 12 98
 e-mail : ramazankocaoglu@gazi.edu.tr



Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek lisans	Gazi Üniversitesi /Bilgisayar Bilimleri	2012
Lisans	Gazi Üniversitesi /Bilgisayar Sistemleri Öğretmenliği	2009
Lise	Anafartalar Anadolu Ticaret Meslek Lisesi	2004

İş Deneyimi

Yıl	Yer	Görev
2017-Halen	Keydata Bilişim Teknolojileri	Ağ ve Güvenlik Direktörü
2015-2017	Verify Veri İletişim Teknolojisi	Ağ Çözümleri Takım Lideri
2014-2015	Adeo Bilişim Danışmanlık Hizmetleri	Ağ ve Bilgi Güvenliği Danışmanı
2010-2013	4S Bilgi Teknolojileri	Ağ ve Bilgi Güvenliği Uzmanı

Yabancı Dil

İngilizce

Yayınlar

Kocaoğlu R., Akcayol M.A. (2011). Kablosuz örgü ağlarda verimlilik analizi ve çözüm önerileri. *Bilişim Teknolojileri Dergisi*, 4(3), 1-12.

Kocaoğlu R., Akcayol M.A. (2012). A scalable routing protocol for wireless mesh networks. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 27(4), 891-899.

Erdem O.A., Kocaoğlu R., (2014). A new Approach for Network Security: Dynamic Intelligent Firewall Architecture. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 29(4), 707-715.

Hobiler

Yüzme, Kitap okumak



GAZİ GELECEKTİR..